ON THE MANAGEMENT OF WIRELESS SENSOR NETWORKS

Linnyer Beatrys Ruiz, Jos´e Marcos S. Nogueira and Antonio A.F. Loureiro

Abstract - During the last decade, there was a great technological advance in the development of smart sensors, powerful processors, and wireless communication protocols that when put together create a wireless sensor network (WSN). A wireless sensor network aims to collect data and, sometimes, control an environment. This kind of network is composed of hundreds to thousands of elements, called sensor nodes, which are often referred as a new class of computer systems, distinguished from the hardware of the past by their ubiquity and their collective analytical skill. The sensor nodes are projected with small dimensions (cm3 or mm3) and this size limitation ends up restraining the their resources, like energy, processor and transceiver capacity. Smart, autonomous, and self-aware: that is the ultimate vision for WSNs. The success of this vision depends fundamentally on the self-management solutions. In this paper we have proposed to use autonomic computing as a paradigm to implement self-managed WSNs a case study, we focus on a class of hierarchical and heterogeneous WSNs where network elements collect data and send them to observers continuously along time. The cost of sending data continuously may lead to a more rapid consumption of the scarce network resources and, thus, shorten the WSN lifetime. However, this is an important kind of WSN and we show that the use of some self-management services proposed by MANNA architecture can promote the productivity of the resource and control the quality of the services provided.
Keywords: Wireless sensors networks management, autonomiccomputing, self-management
Resumo - Durante a ´ultima d´ecada houve um grande avanc¸o tecnol´ogico no desenvolvimento de sensores inteligentes, processadores mais poderosos e protocolos de comunicac¸.aosem _o, que quando agrupados formam uma rede de sensores sem _o (RSSF). Este tipo de rede tem como objetivo coletar dados do ambiente e, algumas vezes, enviar comandos de controle para esse mesmo ambiente. Uma RSSF pode ser composta por centenas a milhares de n´os sensores, que de_nem uma nova classe de elementos computacionais com um hardware distinto do encontrado em outros elementos tradicionais. Estes elementos tendem a ser usados em aplicac¸ .oes ub´_q¨uas e cooperativas. N´os sensores tendem a ser projetados com pequenas dimens.oes (cm3 ou mm3), o que limita os seus recursos como energia, capacidade de processamento e comunicac¸.ao. A vis.ao desejada para RSSFs ´e que tenham capacidade de processar de This work is partially supported by the Brazilian Natioal Research Council (CNPq), process no. 55.2111/2002-3. The authors are with the Federal University of Minas Gerais.(Emails: flinnyer, jmarcos, loureirog@dcc.ufmg.br) forma aut.onoma a partir de eventos percebidos pelos n´os sensores. O sucesso desta vis.ao depende fundamentalmente de soluc¸ .oes de auto-gerenciamento. Neste trabalho n´os propomos usar computac¸.ao auton.omica como um paradigma para implementar uma RSSF auto-gerenciada. Como estudo de caso, n´os estudamos uma RSSF heterog.enea, hier´arquica e com n´os sensores que coletam dados continuamente e os enviam para n´os observadores que est.ao na fronteira entre uma RSSF e uma outra rede qualquer como a Internet. O custo de enviar dados continuamente pode levar a um consumo mais r´apido dos recursos limitados da rede e, assim, diminuir o tempo de vida da RSSF. No entanto, este ´e um tipo importante de RSSF e, neste trabalho, n´os mostramos que o uso de servic¸os de auto-gerenciamento propostos pela arquitetura MANNA pode promover a produtividade dos recursos e controlar a qualidade dos servic¸os providos pela rede.
Palavras-chave: Gerenciamento de redes de sensores sem _o, computac¸.ao auton.omica, auto-gerenciamento

1. INTRODUCTION

Wireless Sensor Networks (WSNs) are an emerging technology that promises unprecedented ability to monitor, instrument, and eventually control the physical world. WSNs, in general, consist of a large number of inexpensive wireless devices (sensor nodes) densely distributed over the region of interest. Sensor nodes are very compact and autonomous, each containing one or more sensor devices, computation and communication capabilities, and power supply. The logical component of a sensor node is the software that runs in its computational unit. Sensor nodes tend to be designed with small dimensions and this size limitation imposes restrictions to the resources of a node, uch as energy, communication and processor capacities. Smart, autonomous, and self-aware: that is the ultimate vision for WSNs. The success of this vision depends fundamentally on the self-management solutions. The task of building and deploying self-management solutions in environments where there will be tens of thousand of network elements with particular features and organization is very complex. This task becomes worse due to the physical restrictions of these unattended sensor nodes. In this paper we show as the MANNA . a WSN management architecture proposed in [?] . can be used to develop self-managed WSNs. Computing systems that can manage themselves given highlevel objectives have been named .autonomic computing. By IBM [?], whereas others prefer more generic, if less evocative, terms such as .self-managing.. But the idea itself most certainly has caught on. As a case study, we focus on a class of hierarchical and heterogeneous WSNs where network elements collect data and send them to observers continuously along time. The cost of sending data continuously may lead to a more rapid consumption of the scarce network resources and, thus, shorten the WSN lifetime. We show that the use of some self-management services proposed by the MANNA architecture can promote the productivity of the resource and control the quality of the services provided. This paper s organized as follows. Section 2 presents a discussion about the management challenges of WSNs. Section discusses the characteristics of a WSN self-managed. Section 4 presents an overview of the MANNA architecture. In Section 5 we describe the self-management services established by the MANNA architecture to provide selfmanagement and in Section 6, we describe the management functions. Section 7 describes the WSN models used to obtain the conditions to perform self-management services and in Section 8 we consolidate all these concepts presenting a case study. Finally, our concluding remarks are given in Section 9.

2. MANAGEMENT CHALLENGES

One of the major goals of network management is to promote productivity of the network resources and maintain the quality of the provided service. However, there are several signi_cant differences in the management of traditional networks and WSNs. In this section we discuss important characteristics of WSNs that make their management different from other networks. WSNs usually work in unattended mode in remote areas and must be able to operate under very dynamic conditions. In some applications, the nodes are dropped over a established area and besides all the planning and care involved in the deposition of the nodes, the initial con_guration can differ from the one initially planned. In unpredictable situations, con_guration errors and even the environment interference may cause the loss of an entire WSN even before it starts to operate. Sensor nodes have strong hardware and software restrictions in terms of processing power, memory capacity, battery lifetime, and communication throughput. In WSNs the main physical restriction is the available energy, since batteries are not, in general, recharged during the operation of a sensor node because of operations in hostile or remote environment and the number of nodes. All activities performed by the node must take into account energy consumption, including management tasks. The number of sensor nodes in WSNs can be several orders of magnitude higher than the number of nodes in an ad hoc network. In general, dense deployment of network elements (sensor nodes) allows greater sensing task and also fault tolerance because of a high level of redundancy. Due to sensor node deployment in environments where they may be lost or destroyed, and in cases where sensor nodes cannot be carefully positioned relative to each other and the environment, a possible strategy to achieve coverage is to deploy a large number of elements. In some contexts, even if the elements are uniformly placed in tri-dimensional space, environmental conditions might be such that the coverage is not uniform due to obstacles and other sources of noise. Another motivation for using a large number of sensor nodes is the case where the incremental cost of deploying a node during initial deployment is much lower than the incremental cost of deploying new nodes or renewing node resource. However, when there are sensor nodes close to each other working on the same sensing event, we can have redundant data generating more traf_c, probably causing interference, increasing the latency of the network, and consuming more energy, which is a precious resource in WSNs. During the operation phase, sensor nodes can be discarded, lost, and stay out of operation temporarily or permanently. It is said that the nodes are disposable. In this network, a failure is not an exception and the topology can change very frequently, even if the nodes are stationary after deployment. WSNs are heavily dependent on the purpose of the application. They are employed in specialized tasks and their nodes cooperate among themselves to perform a larger task (i.e., there is clearly a common goal in the overall network), which may not be the case in a traditional network that are oftendesigned to accommodate a diversity of applications. The WSN management introduces new challenges due to scarce network resources, dynamic topology, traf_c randomness, energy restrictions, and the large amount of network elements. Several aspects of WSNs presented in this section pose new design challenges and research opportunities. In the following, we present an overview of the efforts undertaken with regard to the proposition of a self-management framework for wireless sensor networks. In particular, we establish automatic management services which can be used by a management solution in order to perform self-management.

3. SELF-MANAGED WSN

In this work, we propose that WSNs must be self-managed and robust to changes that occur in the network conditions while maintaining the quality of service. A self-managed WSN must con_gure and recon_gure itself under varying (and in the future, even unpredictable) conditions. System con_guration or .setup. must occur automatically, as well as dynamic adjustments to the current con_guration to best handle changes in the environment. It always looks for ways to optimize its functioning. It will monitor its components and _ne-tune work_ow to achieve predetermined system goals. It must perform something akin to healing . it must be able to recover from routine and extraordinary events that might cause some of its parts to malfunction. The network must be able to discover problems or potential problems, such as uncovered area, and then _nd an alternate way of using resources or recon_guring the system to keep it functioning smoothly. According to [?], a system which has these characteristics can be called an autonomic systems. A self-managed WSN must detect, identify and protect itself against various types of attacks to maintain overall system security and integrity.
A self-managed WSN must know its environment and the context surrounding its activity, and act accordingly. The management entity must _nd and generate rules to perform the best management according to the current network state.
A WSN is a tool for distributed sensing of one or more phenomenon, and reporting the sensed data to one or more observers. A WSN provides services for the observer(s) as well as for itself. It produces, processes and transports application data. In this sense, the network provides *self service* as well. Basic WSN services are sensing, processing, and data dissemination [?]. Thus, we de_ne that in addition to the elements and network management, self-management must also be in charge of QoS aspects namely QoS sensing, QoS processing and QoS dissemination. The *sensing quality* involves sensors calibration, media

interference over the sensor device and the exposure (time, angle and distance between the phenomenon and sensor device). The *processing quality* depends on the robustness and complexity of the algorithms used, and the processor and memory capacity. The*disseminating quality* can be characterized by the latency, the relation bandwidth/delay and by the number of lost messages. The quality of the network service must also be determined by the energy consumption of the executed services with a given quality level. In most WSNs, energy consumption is one of the main design metrics. However, there are situations in which, when certain events occur, the nodes must use the maximum energy available to deliver the data collected from the network. As an example of this situation we can mention WSNs deployed over a havoc of a cave where a user wants to get as much information as it can in as little time as possible. In this kind of application, the extension of the network lifetime is not the goal. However, without proper management mechanisms, the network can suffer from the implosion problem (congestions, collisions, and data losses in the network). In the management solution to be built, there is a tradeoff to be considered: the higher the number of monitored management parameters, the higher the energy consumption and the lower the WSN lifetime. On the other hand, if enough parameter values are not obtained, it may not be possible to manage the network appropriately. In the scope of WSNs, we de_ne that a self-managed WSN has the following capabilities: self-discovery, selfcon _guration, self-organization, self-diagnosis, self-healing self-maintenance, self-optimization, self-protection, selfservice negotiation, self-awareness, self-knowledge, selfsustaining. These capabilities are performed through automatic management services de_ned by the MANNA architecture and de_ned in Section 5.

4. THE MANNA ARCHITECTURE AS SELF-MANAGEMENT SOLUTION PROVIDER

The MANNA architecture [?] is based on the paradigm of self-management and it provides management solutions to different WSNs. The approach used in the MANNA architecture introduces a novel organization for WSN management considering the two well-known management dimensions, namely, functional area management, management levels, and a novel dimension called WSN functionalities (see Figure 1). The traditional anagement dimensions are revisited from a WSN erspective. Considering that WSNs are application-dependent and looking at the characteristics of various WSN applications, _ve main WSN functionalities were established: con_guration, sensing, processing, communication, and maintenance. These functionalities de_nean original dimension for the management, as depicted in the upper part of Figure 1. The intersection of the three planes (dimensions) de_ne a cell. Each cell contains a set of management functions. One or more management functions can _t into one or more cells of the cube.

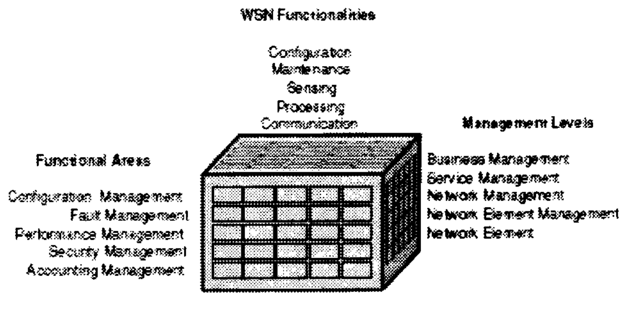


Figure 1. Tri-dimensional organization for WSN management.

The tri-dimensional organization for WSN management must be considered in the de_nition of management services and functions, in the establishment of information models, and in the development of a management application. The use of the management dimensions is a good strategy to deal with complex management situations.
A management service is composed of one or more management functions. Different services can use common functions and the conditions to perform a service or a function come from the WSN models. The WSN models, de_ned by the MANNA architecture, represent a state abstraction of the network and serve as a reference for the self-management. Figure 2 shows the scheme to built a management application from services, functions and models.

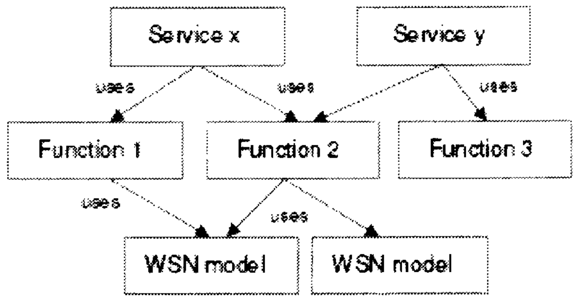


Figure 2. Services, functions and WSN models.

In a self-management application, automatic management services are used, i.e., a set of management functions are automatically performed by a management entity (manager and agent), invoked as a result of the information acquired from one or more maps (WSN models). The location of the management entities and the functions that they can perform are suggested by the functional architecture. The MANNA architecture also proposes two other architectures: physical and information [?]. The adoption of a strategy based on information models and the traditional framework of management functional areas and management levels will make possible management integration in the future.

5. DEFINING MANAGEMENT SERVICES

The de_nition of the management services1 is a task that consists in _nding which activities or functions must be performed, when and with which data they should be performed. As mentioned before, management services are carried out by a set of management functions (see Section 6), and they need to succeed to conclude a given service. The input data for each function is obtained from management information base or in the WSN models (see Section 7). Management services can be clustered and performed by distinct functions in different manners (automatic, semiautomatic, and manual) according to the WSN application. Some of the management services sets are described in the following. In the service de_nitions, the term .entity. means a node, a cluster of nodes or a WSN.

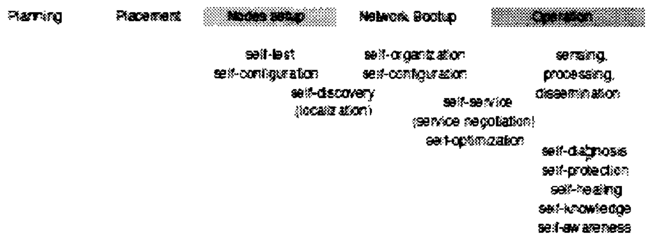


Figure 3. An example of self-management services in WSNlife

Planning. This service involves the design of the network and the decisions about the node architecture, number of nodes, type of deployment, and so on. It is the management service performed before the network boot up time responsible for deciding how the monitoring network is to be placed and performed. Examples of management functions, which can compose this service are: environment requirements acquisition function, monitored area de_nition function, environment monitoring function, node de_nition, number of nodes, and node deployment de_nition function.

Placement. Is the management service that includes all functions related to sensor node deployment on a certain region.

Self-organization. Refers to the management service used to achieve the necessary organizational structures without requiring human intervention. A WSN can be organized in groups of nodes. In this case, it is said to be hierarchical. In other case, it is said to be _at. The ef_ciency of this organizational process can be heavily dependent on the particular deployment of the network and the degree and accuracy of the information that is pre-programmed in the nodes. The self-organization service is composed of initialization routines, network discovery routines, node type announce-ment, program/command injection/exchange, topology learning and position determination routines, nodes scheduling, initial traf_c determination routines, routing routines, medium control access routines, network time distribution routines, and dynamic connection establishment/termination routines. Self-organization means to adapt dynamically to environmental conditions and states of the network variation in order to maintain its operation. The characteristics of sensor nodes necessary for creating self-organizing sensor networks are agility, self-awareness, self-con_gurability, and autonomy. Sensor nodes with these features will have capabilities for self-assembling impromptu network degradation, mobility of sensor nodes, and changes in task and network requirements. Building self-organizing WSNs is dif_cult because of the following main reasons: many different types of sensors with a range of capabilities must be developed with different application requirements; the use of data-centric network protocols (such as directed diffusion); the network must be extensible to new types of sensor nodes and services; and the network must react rapidly to changes in the topology, task, degradation, and mobility. **Node setup**. Upon power up, a node will perform a number of initialization routines, such as internal node self-test health status determination, and built-in calibration. It will also launch any procedures that have been pre-programmed to re_ect speci_c mission requirements and expectations. **Self-con_guration.** Is the management service that changes the parameters of con_guration to adapt itself dynamically to the changing conditions or states of the network. It congures and recon_gures itself under varying (and in the future, even unpredictable) conditions. System con_guration or .setup. must occur automatically, as well as dynamic adjustments to the current con_guration to best handle changing environments. **Self-diagnosis**. Is the management service that quali_es the network to monitor itself and _nd faulty or unavailable nodes. **Self-protection**. Is the management service that anticipates, detects, identi_es and protects the entity against threats (internal and external, accidental and malicious). When an attack happens, these services perform intrusion detection routines to reach security and safe states. Self-protection features include con_dentiality, integrity, reliability, disposability, privacy, authenticity, and integrity.

Self-healing. Is the management service that prevents disruptions or that acts to recover the network or the node after the self-diagnostic (if possible). It enables the entity to recover from problems that might have happened. It must be able to discover potential problems and then _nd an alternate way of using resources or recon_guring the entity to keep in normal operation. **Self-optimization**. Is the management service that tunes resources and balances tasks to maximize the use of resources, minimizes latency, and maintains the quality of service. An entity always looks for ways to optimize its job. **Self-service**. Is the management service that enables an entity to provide sensing, processing, and disseminating services, anticipating the optimized resources needed while keeping its complexity hidden. It must marshal resources to shrink the gap between the application business or service goals (QoS sensing, QoS processing, and QoS disseminating), and the

1Note that the term management service is different from the service management functional area.

implementation necessary to achieve these goals. **Self-awareness**. Is the management service that allows the entity to know its environment and the context surrounding its activity, and act accordingly. It will _nd and generate rules for best interacting with neighboring entities. It will tap available resources, even negotiate their under-utilized elements used by other entities, changing both itself and its environment in the process . in a word, adapting. **Self-knowledge**. Is the management service that quali_es an entity to .know itself.. For example, an entity that governs itself must know what are its components, current state, ultimate capacity, and all connections to other entities. It will need to know the extent to which its resources can be borrowed, lent, or shared. **Self-sustaining**. Is the management service that uses budget schemes to prevent energy waste and promote rational use of energy in order to survive. **Self-maintenance.** Is the management service that enables an entity to monitor its components and _ne-tune itself to achieve pre-determined goals of an entity. One of the main examples of the maintenance services is .coverage area maintenance management service., which uses the density control function to identify the nodes that can be administratively stay out of service (redundant node) in order to reduce congestion, collision and energy waste. The network density self-management depends on the application requirements. The self-managed WSN has the _exibility to negotiate services, that is, of meeting the performance demands by controlling the reporting rate of the sensors, controlling the virtual topology of the network by scheduling of nodes, or optimizing the collective reduction communication operation by data aggregation. In this sense, the self-managed WSN performs service negotiation. An approach to perform this service is used in this paper as a case study. Other examples of management services in this set are: QoS maintenance, mobile management, scheduling task, denial of service avoid, key management and differentiation of services.

6. DEFINING MANAGEMENT FUNCTIONS

The management functions represent the lowest granularity of functional portions of a management service, as perceived by users. A scheme to design management functions consists in dealing with each management functional area and each management level considering the functional model of the network and establishing what are the management tasks found in the intersection of the three dimensions (see Figure 4).

As result, a partial list of the management functions, in no particular order, is given in the following. **Environment requirements acquisition function**: consists in obtaining requisites about the environmental conditions of the area to be monitored. **Monitored area de_nition function**: consists in establishing

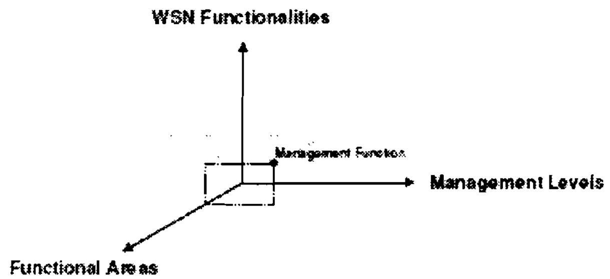


Figure 4. Intersection of the management dimensions the size and the form of the region to be monitored.

Environment monitoring function: consists in supervising variations in the environment that extrapolate de_ned thresholds. **Node de_nition function**: consists in de_ning the node architectures to be used in the network to perform the de_ned services. **Number of nodes de_nition function**: consists in de_ning the number of nodes to be deployed. **Node deployment de_nition function**: consists in determining the location and the way in which the nodes will be placed in the monitored region. We developed an ef_cient incremental sensor network deployment algorithm in [?]. **Network operating parameters con_guration function**: consists in attributing values to the network parameters and to the nodes. **Node deployment function**: after the de_nition, the nodes can be deposited in a random or uniform-distributed environment. **Topology map generation function**: consists in discovering the topology of the network. The topology describes the connection that may exist and expresses the relationships among the sets of nodes. **Network connectivity discovery function**: consists in discovering the connectivity of the network. The connectivity represents the association between two network points at a given instant. **Correlation discovery function**: consists in discovering the relationship between a compound object and its immediate components. **Cooperation discovery function**: consists in obtaining the cooperating relations. **Synchronization function**: consists in the execution of synchronizing functionalities that may be used in functions such as cryptography. **Energy map generation function**: consists in obtaining the energy map of the network. **Node density calculation function**: consists in discovering the quantity of nodes per monitored area. We developed a management service to control the network density in [?]. **Network coverage area de_nition function**: consists in strategic planning for the establishment of the covered network area, considering the area type (internal or external), dimensions, environmental conditions, conditions of node disposition, and so on. **User interface function**: consists in executing functions to interface with observers.**Node programming**: consists in programming the sensor nodes to perform application tasks, power management, mobile management, and so on. **Self-test function**: consists in running tests done by the nodes themselves. **Node localization function**: consists in discovering the node location. This function allows the utilization of different methods of global or relative localization. **Node operating state control function**: due to the different activities and the energy level in the nodes and in the network, the network nodes may present different operational states: normal, major, minor, critical, and inactive. **Node administrative state control function**: there may bemoments in which it is desirable for a node to change its administrative state for the interest of the application, for instance in the case of two sensor nodes presenting an intersection in the covered area. **Node usage state control function**: consists in controlling which elements of the node are in use. **Energy level discovery function**: each node may notify its energy residual level. **Leader election function:** consists in algorithms to choose or elect leaders (cluster-heads) such that each sensor node will be associated with at least one cluster-head as its leader in a hierarchical WSN. **Invitation to form cluster**: the leader nodes send invitation to other nodes to form a cluster. **Listening for invitation**: the nodes listen to other nodes for invitations to form a cluster. **Response to invitation**: once a node hears an invitation to join the network, it transmits a response. It is possible that multiple nodes will hear the same invitation and then they will be part of the same cluster. **Calibration**: typical traditional single-sensor calibration relies on providing a speci_c stimulus with a known result, thus creating direct input-to-output mappings. The calibration for any sensor is subject to speci_c ranges and operating conditions, which are reported in the speci_cations of the sensor. **Power management**: consists in a plane to manage how a sensor node uses its power. **Mobility management**: consists in detecting, planning, running, and registering the movement of sensor nodes. **Task management**: consists in balancing and scheduling the sensing, processing and disseminating tasks given in a specific region. Task management can include micro tasks of the management. **Coverage area supervision function**: consists in supervising the alterations in form and size of the monitored area. **Priority of action de_nition function**: allows the establishment of priorities for operational actions depending on the state of the network. **Management operation schedule function**: makes possible the establishment of a plan for the managing operations. **QoS Monitoring Function**: consists in monitoring the quality of sensing, processing, and disseminating services. We de_ned some functions that allow us to obtain characteristics, which can determine the ef_ciency and effectiveness of a WSN. Some of these quantitative functions de_ned to obtain parameters are presented in [?]: **Network settle time function**: consists in obtaining the time required for a collection of nodes to automatically organi e itself and transmit the _rst message reliably. **Network join time function**: consists in acquiring the time necessary for an entering node or group of nodes to become integrated into an ad hoc network. **Network depart time**: consists in obtaining the time required for the network to recognize the loss of one or more nodes, and reorganize itself to route around the departed nodes. **Network recovery time function**: consists in obtaining the time required for a collapsed portion of the network (due to traf_c overload or node failures) to become functional again once the load is reduced or the nodes become operational. **Frequency of updates (overhead) function**: consists in de_ning the number of control packets required in a given period of time to maintain formal network operation. **Memory requirement function**: consists in calculating the requisites of storage space in bytes, including routing tables and other management tables. **Network scalability function**: consists in _nding the network threshold, which is the number of nodes the network may escalate and con_dently preserve the communication. **Energy consumption per task function**: consists in discovering or predicting the energy consumption rate per task. The distributed management architecture MANNA is based on two paradigms: policy-based management and autonomic management. Also, the performance of WSNs and the management application depends on the routing and medium access control of the underlying network. Thus, the qualitative features [?] to de_ne policies regardless of the application involves: **Knowledge of node locations**: Does the routing algorithm require local or global knowledge of the network? **Effect of topology changes**: Does the routing algorithm need complete restructuring or only incremental updates? **Adaptation on radio communication environment**: Do nodes use estimated knowledge of fading, shadowing, or multiuser interference on links in their routing decisions? **Power consciousness**: Does the network employ a routing mechanism that considers the remaining energy of nodes? **Single or multichannel**: Does the routing algorithm use a separate control channel? In some applications, multichannel performance may cause the network to be vulnerable to countermeasures. **Bidirectional and unidirectional links**: Does the routing algorithm perform ef_ciently on unidirectional links, e.g., if bidirectional links become unidirectional? **Preservation of network security**: Do routing and MAC layer policies support the survivability of the network, in terms of low probability of detection, low probability of interception, and security? **QoS routing and handling of priority messages**: Does the routing algorithm support priority messaging and reduction of the latency for delay sensitive real-time traf_c? In the majority of the management applications, the MANNA architecture uses automatic services and functions performed by a management entity invoked as a result of information acquired from a WSN model. Management services and functions can also be semi-automatic, when performed by an observer assisted by a software system that provides a network model or invoked by a management system, and manual, when performed outside the management system. Six possible states are de_ned for a function: ready (when the necessary conditions to carry out a function are satis_ed); not-ready (when the necessary conditions to carry out a function are not met); running (when the function is being performed); done (when the function performed well); cancelled (when a cancellation occurs); failed (when a failure occurs during function execution). The above management function list will be helpful in the development of self-managed WSNs, as well as WSN applications.

7. DEFINING WSN MODELS

In WSN management, there are two kinds of management information: static and dynamic. Static management information describes the con_guration of services, network and network elements. Dynamic management information is described by WSN models and has to be updated frequently. It represents the network states. In a WSN, the network conditions may vary dramatically along the time. In this case, the use of models established by MANNA is of fundamental importance for the management, although its updating cycle can be extremely dynamic and complex. Based on the information obtained with these models, services and functions are carried out according to management policies. In the following, some network models are presented. They always represent dynamical aspects of the network. Some WSN models can be obtained from the combination of other models using management information stored in Management Information Base (MIB). In [?], we proposed a solution to obtain WSN models.**Network topology map**. It represents the topology map and the reachability of the network; **Residual energy**. It represents the remaining energy in a node or in a network. Using the energy map, the management application can determine if any part of the network is about to suffer system failures in the near future due to depletion of energy. We developed a fault-tolerant approach to obtain energy map in [?]; **Sensing coverage area map**. It describes the actual sensing coverage map of the sensor elements;**Communication coverage area map**. It describes the actual communication coverage map from the range of transceivers; **Cost map**. It represents the cost of energy necessary for maintaining the desired performance levels; **Production map**. It represents the nodes that are producing and delivering their data. A production map was developed and used in [?]; **Usage standard map**. It represents the activity of the network. It can be delimited for a period of time, for quantity of the data transmitted to each sensor unit, or for the number of movements made by the target; **Dependence model**. It represents the functional dependency that exists among the nodes; **Structural model**. It represents the aggregation and connectivity relations among network elements; **Cooperational model**. It represents relations of interaction among network entities; **Audit map**. It represents records, which allow the veri_cation of whether a security violation is happening or has already happened; **Coverage area map**. There are some possibilities to determine the coverage area map using the sensing and communication maps, as shown in Figure 5, when considering sensor range and radio range: sensor range greater than, less than, or equal to radio range. We developed an approach to obtain sensing coverage map and communication coverage map in [?].

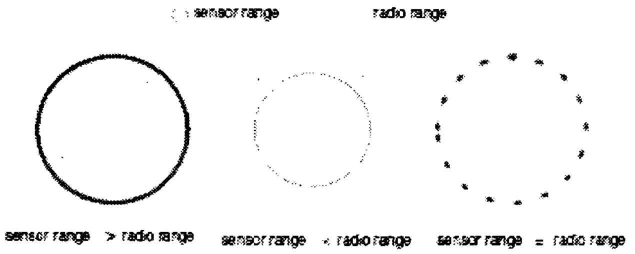


Figure 5. Sensor and radio range possibilities.

8. PUTTING IT ALL TOGETHER

The management application to be build as a case study depends on the kind of application being monitored. In our study, the application that runs in the WSN monitors some parameters that de_ne the air quality such as temperature and carbon monoxide (CO). In this case, the service management must be used in determining how, when and where the application data was produced. The con_guration (in terms of the sensor capabilities, number of sensors, density, distribution, self-organization, self-optimizing, and data dissemination) plays a signi_cant role in determining the performance of the network. As such, the performance of the network and provided service are best measured in terms of meeting the accuracy and delay requirements de_ned by the observer, as well as consumed energy. Additional performance metrics include coverage area, exposure, goodput, cost of the sensors, scalability, and produced data quality. We consider the coverage area and accuracy as other metrics relevant in data delivery. Meguerdichian [?] de_nes *coverage area* as a meas re of QoS for WSN. In the worst-case coverage, attempts are made to quantify the quality of service by _nding areas of low observability from sensor nodes and detecting breach regions. In the best-case coverage, the management application has to _nd areas of high observability from sensors and identify the highest accuracy. In this case study, we design a network with a high node density and use the density control management function to turn off redundant nodes. The provision of QoS relies on resource reservation. When the active node goes out of service (due to operational problems), the management application active redundant node, de_ning a sort of resource reservation scheme. In case of a low density, the network coverage area can be compromised affecting the quality of the service. We consider that we are applying resource reservation. When a WSN does not have a management solution, its sensor nodes collect, and disseminate this data through the network. Considering that the network is dense, it is likely to have congestion and collision, consequently, higher delay, message loss, and wasting energy. In self-managed WSNs, after the nodes are dropped, they wake up, perform a self-test, _nd out their localization (selfdiscovery) and monitor their energy level, usage state, and administrative state (self-knowledge). These activities are performed by automatic management services at network element level. The management challenge is to perform this task without adversely consuming network resources. Once the nodes _nd out their location, they can organize themselves in groups (self-organization). The self-organizing is a management service performed at network management level (see Figure 1). The wireless sensor nodes must adapt themselves to changes in the environmental conditions (selfawareness). At service management level, a WSN uses its own transport service, i.e., the network produces the information and transports it to its destiny (self-service), maintaining the service quality level established (self-maintenance) and protecting against attacks (self-protection).

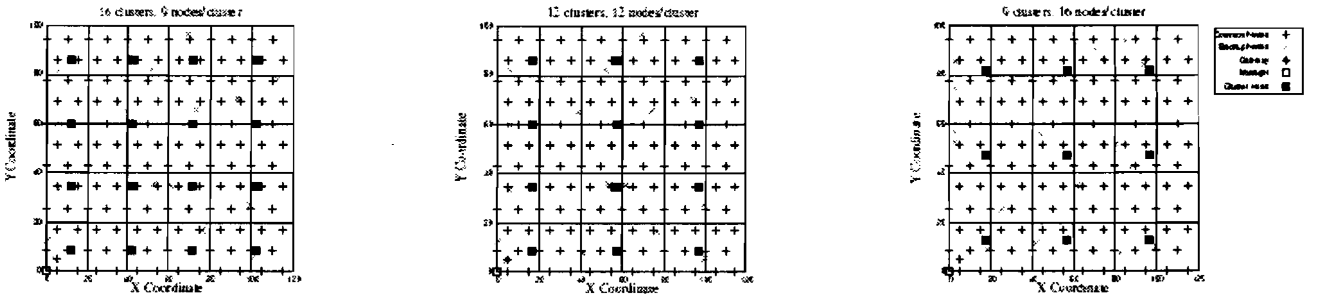
ON THE MANAGEMENT OF WIRELESS SENSOR NETWORKS
Linnyer Beatrys Ruiz, Jos´e Marcos S. Nogueira and Antonio A.F. Loureiro

8.1 BUILDING A SELF-MANAGEMENT WSN

In our case study, we adopt the external manager, which has a global vision of the network and can perform complex tasks that would not be possible inside the network. The performance management in the *service level* involves how, where and when the data was produced. The performance management in the *network level* involves accuracy, goodput and latency. In the *network element level*, the performance is affected by the accuracy of the sensing hardware (transducer), size of memory (the buffering space), battery capacity, capabilities of the embedded processor (determinate the level of optimization that is possible at the sensors without introducing excessive loss of power or intolerable delay), and characteristics of the transceiver (determinate the transmission range of the network and the capacity of the transmission channel). The proposed management application is divided into two phases: installation and operation. The installation phase occurs as soon as the nodes are deployed in the network. In this phase, each node _nds out its position in the area and reports it to the agent located in the cluster-head. The agent aggregates this information and sends a POSITION TRAP of its location to the manager. The common nodes also inform their energy level that the agent aggregates in an ENERGY TRAP, and sent to the manager. The management application builds automatically all neededWSN models based on both local information and data sent by the agents, i.e., the *WSN topology map* and *the WSN energy map*. These two models are used to build the *WSN coverage area map*, which the manager uses to monitor the sensing and communication coverage area, and to calculate the density of the network. The MANNA architecture proposes a *coverage area maintenance service* and a *density control function*, which can reduce system overall energy consumption, therefore increasing the system lifetime,by turning off some redundant nodes in dense networks. This service preserves the sensing coverage with minimum sensing hole and maintains the system reliability. To execute this service management, the manager sends a SET operation to change the administrative state value of the node attribute and set a wake up interval. When detecting minimal levels of energy or uncovered areas, the management application activates the backup nodes. The management application also implements a *network operating parameters con-_guration service*. The manager consults the topology map and adjusts the transmission power (communication range) of the cluster-heads. The nearest cluster-heads from the BS will have a reduced range, saving energy. In some contexts of WSNs, applications are less tolerant concerning some metric related to the data that _ows from source nodes to a cluster-head, and from a cluster-head to the base station, called SENSOR.REPORT. For example, the loss of a single message associated with a cluster-head would render imprecise maps. In our experiment, we evaluate the goodput, latency and accuracy of the management and application data, i.e., performance evaluation. In the operation phase, while the sensor nodes are performing their functions, i.e., collecting and sending temperature and carbon monoxide level data, management activities take place. Among them, energy level monitoring plays a central role. Each node checks its energy level and sends a message to the agent whenever there is an operational state change. This information is transmitted to the manager through a ENERGY TRAP. Any information the agent receives is recorded in its Management Information Base. The manager can, then, recalculate the energy and topology maps, as well as the coverage area, which characterizes the coverage area maintenance service. When a common node has the critical energy level (less than 10%) it sends a DELETE TRAP, which is directly sent to manager (with no aggregation). The manager receives a DELETE TRAP, it tries to activate backup nodes. The management application uses the production map to manage the quality of service. In a continuous application, when the management application stops receiving SENSOR. REPORTs from a given node, this may be an indication of a problem. Thus, the manager consults the energy map to verify if it has residual energy. If so, the manager detects a production problem and sends a QoS noti_cation to the observer. In this way, the MANNA architecture provides erformance monitoring in continuous WSN with associated cost only to TRAPs and some SETs sent because the management takes advantages of the features of the network to obtain management information indirectly.

8.2 SIMULATION APPROACH

In our application, the carbon monoxide level and temperature are the monitoring objects. The nodes sense the phenomenons and disseminate the data continuously along the time. In order to simulate the phenomenom behavior of the environment, random numbers were generated considering a standard deviation of 1, from a temperature interval of 22_C to 32_C and carbon monoxide (CO) level between 30:000_g/m3 and 50:000_ g/m3. We consider a regular deployment in three distinct types of network hierarchical organization. Our goals were to evaluate the impact of the network configuration over the performance and services, and to evaluate the impact of the management application over the WSN latency, goodput and energy consumption. For this, six scenarios were de_ned and simulated in respect to distinct network con_gurations and management application (see Figure 6) : **Scenario 1**: 16 clusters, 9 common nodes per cluster,
10% redundancy, without management.
_ **Scenario 2**: 12 clusters, 12 common nodes per cluster,
10% redundancy, without management.
_ **Scenario 3**: 9 clusters, 16 common nodes per cluster,
10% redundancy, without management.
_ **Scenario 4**: 16 clusters, 9 common nodes per cluster,
10% redundancy, with management.
_ **Scenario 5**: 12 clusters, 12 common nodes per cluster,
10% redundancy, with management.
_ **Scenario 6**: 9 clusters, 16 common nodes per cluster,
10% redundancy, with management.
We have de_ned a WSN application and some management functions, as mentioned before, and evaluated these scenarios using the Network Simulator (ns-2) [?], version 2.1b9a. The results presented have a 95% condence



(a) Scenarios 1 and 4 (b) Scenarios 2 and 5 (c) Scenarios 3 and 6

interval. In the scenarios evaluated, we considered the following variables:

Network. It comprises 144 common nodes that are distributed in a uniform manner upon the monitored area (115m _ 95m) and there is more 10% redundant common nodes. The nodes are organized in clusters. Each cluster will have a cluster-head. Protocols: IEEE.802.11, AODV, SNMP, and MNMP.

Common nodes [?]. Bandwidth: 50kbps, transmission power: 0.036J, reception power: 0.0054J, communication range: 40m, processing power (active: 0.0165J, idle: 0.0048J, sleep: 0.00006J), sensing power (temperaturesensor: 0.0006J, carbon monoxide sensor: 0.001J),sensor range: 6m, battery capacity: 0.8J, without mobility.

Cluster-head nodes [?, ?]. Bandwidth: 50kbps, transmission power: 1.176J, reception power: 0.588J, communication range: 140m, processing power (active: 0.0165J, idle: 0.0048J, sleep: 0.00006J), battery capacity: 40J, without mobility. We are interested in service level performance, conventional network performance metrics, such as throughput, are of secondary interest.

8.3 SIMULATION RESULTS

One of the major goals of network management is to promote productivity of the network resources and maintain the quality of the provided service. In this section, we investigate the effect (tradeoff) of the MANNA management architecture on the performance of a WSN. We present the results for the following performance metrics: accuracy, delay, energy efficient, and goodput. In order to investigate in_uence of the con_guration, we conducted all experiments with three types of hierarchical and heterogeneousWSNs con_gurations, with and without management.

Accuracy. The accuracy of a measurement at a *network elementis* speci_c to the physical transducer, the nature of the phenomenon, and the exposure. The accuracy at a *network level* depends on the delay in the data delivery due to network congestion, the duty cycle of the sensors, or aggregation processing of sampling data. The accuracy at a *service level* depends on the metric chosen by the application for establishing the coverage area and amount of energy to be spent in gathering and disseminating data. At the observer, it is likely that multiple samples will be received from the different sensor nodes, producing data quality. For the de_ned application, depending on the network latency and uncovered area percentage, the data received by the observer may be of no valueand should be discarded. It is expected that increasing the number of sensors per cluster it results in better accuracy and lifetime. Since there are more sensors in a position to report on the phenomenon, the accuracy of the sensing gets better. The available energy within the network increases and the additional sensor density offer the potential for a better connected network with more efficient paths between the sensor nodes and the observer(s). Nevertheless, increasing the number of sensor nodes per cluster implies in a higher number of nodes disseminating their results per time unit. The problem can be viewed in terms of collision and congestion. For the continuous update reporting model (all sensors report data continuously), we study the effect of the number of clusters and number of nodes per cluster. Figures 7 and 8 show the coverage area map (WSN model defined by MANNA) at instant 31s and 121s of simulation. There are three types of observability areas: uncovered, covered, and covered intersection (highest accuracy). The uncovered area in Figure 7(a) and (b) is significantly smaller than in the Figure 7(c) at 31s. There are more intersection areas in Figure 7(a). At 121s, the network is not available and the small covered areas (see Figure 8) are related to the backup nodes that were activated. If there were more available backup nodes, the management application could promote the extension of the network lifetime. This difference can also be observed in the Figure 9, which shows the data delivery for the produced data. At instant 31s of simulation, the scenario 5 has the better delivery rate (91.94%) and the scenario 4 has 90.21% whereas in the scenario 6, only 66% of the application data is delivered. In instant 50s, the scenario 6 has a percentage of data delivery of 92.21% due to late messages that arrived. After 100s of simulation, most of the common nodes has already gone out of service permanently due to their energy level. There is about 8% of nodes producing, which are the backup nodes activated.

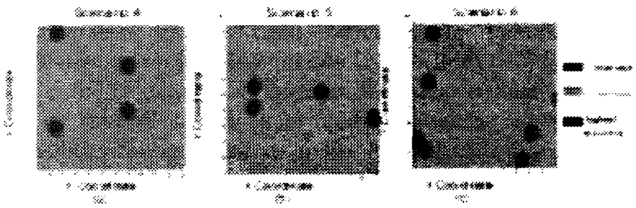


Figure 8. Coverage area map at time simulation 121s for the scenarios 4, 5 and 6.

The question is: Why the coverage area and data delivery are different at instant 31s in the Figures 7(a), (b) and (c)? We can notice that the number of clusters and the cluster size have influence in this metric since the amount of generated messages by the common nodes and the protocol stack are the same. Figure 9 also shows the rate of late data packets per simulation time. This rate for scenario 6 is quite greater than the value for scenarios 4 and 5, because of the higher number of common nodes per cluster in the former. The delay varies with the number of nodes changing messages, i.e., the transmission among neighboring nodes interferes with each other.

Delay. We observed the effect of increasing the number of sensor nodes per cluster on the data production (delivery) and coverage area. Now, we study the reason why the coverage area map and data delivery is different for scenarios 4, 5 and 6. Observing Figure 10, two kinds of delay can be considered for the purpose of analysis, which are the message delay between the agents (cluster-heads) and the manager (see Figure 10(a)), and the communication delay between the agents (cluster-heads) and the common nodes (see Figure 10(b)). In Figure 10(a), the delay of scenario 1 (9 nodes per cluster) is higher because there are more cluster-heads disputing the physical medium. Using management, the performance is better because of the configuration of the transmission power of the cluster-heads, relating it with their distances from the BS. The nearest cluster-head from the BS will have a reduced range, saving energy. In Figure 10(b), the delay acts in a similar way for all the scenarios. The delay with management is higher because of the messages sent by the nodes to inform their positions, in the installation phase and residual energy, along the operating phase. At the end of simulation, the delay is a bit higher too, once the nodes that have a critical level of energy send a DELETE TRAP. The agent does not make an aggregation of these TRAPs. In this case it considers this information with a high priority and forwards it directly to the manager. Without the aggregation service, the number of disseminated messages is quite high, resulting in collision and loss. Due to the configuration characteristics, there are scenarios in which the cluster-heads process a greater number of messages. In performance management there is a trade-off to be considered: the highest the number of managed parameters, the highest the management cost (Figure 10(c)). On the other hand, if parameter values are not obtained, it may be not possible to manage the network appropriately.

Goodput. It is the ratio of the total number of packets receivedby the observer to the total number of packets sent by all sensors over a period of time. Figure 11(a) shows that the message loss for scenarios 1, 2, and 3 are smaller than the values for scenarios 4, 5, and 6. The difference is 2.7% for scenarios 1 and 4, and 1.8% for scenarios 2 and 5. For scenarios 3 and 6, the difference of the lost messages is 7%. In this case, it is possible to notice that with management there is a higher message loss than without it. This difference is due to largest number of produced messages by management application (SETs and TRAPs) and the bidirectional flow (from agents to manager and from manager to agents), which is not the case in the scenarios without management. Besides, in the scenarios with management, there are three entities producing and sending data, manager, agents and common nodes. Figure 11(b) shows the total number of lost messages for the common nodes. The nodes had to transmit more messages with the management, resulting in a bigger number of messages that congested the physical medium, causing messages to be lost. For the scenario 3, the 16 common nodes per cluster without management cause more collisions than scenarios 1, 2, 4, 5, and 6. In Figure 11(c), it is observed that the scenarios with a large number of nodes per cluster, and, consequently, a small number of agent nodes, have a higher total message loss in respect to the number of messages lost inside the clusters. We can notice that management contributes only a small increase in the total of lost messages (scenarios 4 and 5) but concerning the scenarios 3 and 6, the management contribute to reduce the total lost messages. We can also notice that the introduction of the management has little impact on thismetric.

Energy. It is a critical resource in a WSN. In Figure 12(a), we show that the management saved energy. In Figure 12(b), in terms of energy consumption of common nodes, the management consumed more energy. In Figure 12(c), we observe that the management accomplishes its purposes, contributing to prolong the network lifetime. Comparing the graphics in Figure 12, we observe that the management of the configurable parameters, promote the network productivity, reducing the energy consumption of the cluster-heads. Considering the common nodes, the energy consumption is distributed in a uniform manner in respect to the application network characteristics. Scenarios 4, 5 and 6, which use the management, have some nodes spread along the area with more residual energy than the others. They are the backup nodes and again, if the application had more nodes like these ones, the network lifetime could be extended.

9. CONCLUSION

Environmental monitoring represents an important class for applications in wireless sensor networks. Many kinds of observers are interested in the sensor data, like public and private companies. Therefore, the WSNs must provide the data of interest in a confidence-inspiring manner. Management of WSNs is a new research area that only recently started to receive attention from the research community. In this sense, this work presents a contribution to the field, since it proposes the service management using the MANNA architecture, which is based on traditional framework of functional areas and management levels. The adoption of this strategy will allow management integration in the future. In our experiments, we were able to build the models for the WSN topology map, WSN energy map, WSN coverage area map, cost map, and WSN production map. These models are important in different applications specified and designed for WSNs. Probably the fundamental issues about management of WSN are concerned on how the management application promotes resource productivity and quality of the services. Nevertheless, an important aspect is to verify the impact of the management services over the WSN lifetime, latency, goodput and coverage area. The important point that needs to be stressed is that the introduction of the management services in our experiments did not affect the network behavior considerably. The management reduced the consumption of total energy, although it increased the number of lost messages and delay in the common nodes for some scenarios. Of course, there is a cost associated with the network management and, at the end, the benefits brought by this solution may outweigh the cost paid. In agreement with intuition, the results show that increasing the cluster density can result in a higher accuracy, butonly if the sensing traffic is kept below the network capacity. A specific WSN protocol stack could be used to make the network behavior and, consequently, the simulation more adequate. ther management services, management functions and management types defined by the MANNA architecture can be implemented. We understand that the framework proposed and the list of management services is a relevant contribution for the field, once there was not in literature any proposal related to the theme of WSN self-management. This work has also discussed the management challenges for WSNs and proposed that WSNs can perform autonomic computing. A self-management solution depends on the feature of the network. There are WSNs in which only a few management services can be implemented. In other cases, the selfmanagement solution cannot be performed because of restrictions in the computation and resources. The MANNA architecture provides a framework with this flexibility. The self-management solution can be obtained from the composition of the management services and the definition of management policies which can be performed through a centralized, distributed, and hierarchical approaches. We have proposed some work [Ruiz et al., 2004, Ruiz, 2003, Vieira et al., 2003a, Silva et al., 2003, Siqueira et al., 2004, Vieira et al., 2004] using the management services and functions defined by the MANNA architecture, as well as WSN models. The results demonstrate that the self-management can promote the productivity of the resources and control the quality of the provided services. WSNs promise several advantages over traditional sensing methods in many ways: better coverage, higher resolution, fault tolerance, and robustness. The use of the selfmanagement paradigm has also shown itself to be adequate for the specific features of the WSNs. An application that continuously performs data sensing, processing, and dissemination was defined as a case study. Some management services and functions were chosen to evaluate our solution. The interconnection of sensors through wireless communication networks, with the goal of performing a larger sensing task, encourage several novel and existing applications such as environmental monitoring, health care, infrastructure anagement, public safety, medical, home and office security, and transportation [Badrinath et al., 2000, Lindsey et al., 2001, Meguerdichian et al., 2001]. The ad hoc nature and deploy–and–leave vision make them it even more attractive in military applications and other risk-associated applications such as catastrophe, toxic zones, and disas ter [Estrin et al., 2000]. Wireless sensor networks will also play a key role in pervasive computing where computing devices and people are connected to the Internet.

REFERENCES

[mic, 2002] (2002). The commercialization of microsensor motes. Available in http://www.sensorsmag.com.
[WIN, 2002] (2002). Wireless Integrated Network Sensors (WINS). Available in http://www.janet.ucla.edu/wins/.
[Badrinath et al., 2000] Badrinath, B. R., Srivastava, M., Mills, K., Scholtz, J., and Sollins, K. (2000). Special issue on smart spaces and environments. *IEEE Personal Communications.*
[Estrin et al., 2000] Estrin, D., Govindan, R., and Heidemann, J. (2000). Embedding the internet. *Communications of the ACM*, 43(5):39–41.
[IBM, 2003] IBM (2003). Autonomic Computing – Creating Self-maning Computing Systems. Dispon´ıvel em http://www-3.ibm.com/autonomic/index.shtml.
[Lindsey et al., 2001] Lindsey, S., Raghavendra, C., and Sivalingam, K. (2001). Data gathering in sensor networks using the energy delay metric. In *International Workshop on Parallel and Distributed Computing: Issues in Wireless Networks and Mobile Computing.*
[Meguerdichian et al., 2001] Meguerdichian, S., Koushanfar, F., Potkonjak, M., and Srivastava, M. B. (2001). Coverage problems in wireless ad hoc sensor networks. In *IEEE INFOCOM - Annual Joint Conference of the IEEE Computer and Communications Societies*, pages 1380–1387. [Ruiz, 2003] Ruiz, L. B. (2003). *MANNA: A Management Architecture for Wireless Sensor Networks.* Ph.d. thesis, Federal University of Minas Gerais. [Ruiz et al., 2003] Ruiz, L. B., Braga, T. R., Silva, F., Nogueira, J. M. S., and Loureiro, A. A. (2003). Service management for wireless sensor networks. *IEEE LANOMS – Latin American Network Operations and Management Symposium*, pages 55–62. [Ruiz et al., 2004] Ruiz, L. B., Fabr´ıcio, Braga, T. R., Silva, Nogueira, J. M. S., and Loureiro, A. A. (2004). On impact of management on wireless sensor networks. *IEEE NOMS – Network Operations and Management Symposium*, pages 657–670. [Silva et al., 2003] Silva, A. P. R., Teixeira, F., Lage, R., Ruiz, L. B., Loureiro, A. A., and Nogueira, J. M. (2003). Using distributed snapshot algorithm in wireless sensor networks. *The 9th IEEE Workshop on Future Trends of Distributed Computing Systems*, pages 31–27.
[Simulator,] Simulator, N.-. N. Available inhttp://www.isi.edu/nsnam/ns, July 2003. [Siqueira et al., 2004] Siqueira, I. G., Ruiz, L. B., Loureiro, A. A. F., and Nogueira, J. M. S. (2004). A management service for density control in wireless sensor networks (in portuguese). In *Proceedings of the 22nd Brazilian Symposium on Computer Networks (SBRC'04)*, pages 249–262, Gramado, RS, Brazil. ISBN 85-88442-80-9. [Subbarao, 1999] Subbarao, M.W. (1999). Ad hoc networking critical features and performance metrics. Technical report,Wireless Communications Technology Group, NIST. [Vieira et al., 2004] Vieira, L. F. M., Vieira, M. A. M., Ruiz, L. B., Loureiro, A. A. F., da Silva Jr., D. C., and Fernandes, A. O. (2004). Efficient incremental sensor network deployment algorithm. In *Proceedings of 22nd Brazilian Symposium on Computer Networks (SBRC'04)*, pages 3–14, Gramado, RS, Brazil. ISBN: 85-88442-81-7. [Vieira et al., 2003a] Vieira, M. A., Vieira, L. F., Ruiz, L. B., , Loureiro, A. A., Fernandes, A. O., and Nogueira, J. M. S. (2003a). Scheduling nodes in wireless sensor network: A voronoi approach. *IEEE LCN – Local Computer Network*, pages 423–429. [Vieira et al., 2003b] Vieira, M. A., Vieira, L. F., Ruiz, L. B., Loureiro, A. A., , and Fernandes, A. O. (2003b). Como obter o mapa de energia em redes de sensores sem fio? uma abordagem tolerante a falhas. *V WCSF - Workshop de Comunicac¸ ˜ao Sem Fio e Computac¸ ˜ao M´ovel*, pages 183–189.

Linnyer Beatrys Ruiz holds a B.Sc. degree in Computer Engineering from Pontifical Catholic University of Paran´a (PUCPR), a M.Sc. degree in Electrical Engineering and Industrial Information form the Federal Center of Technological Education of Paran´a (CEFETPR), and a Ph.D. degree in Computer Science from UFMG, Brazil. She held a post-doctoral position at the UFMG, 2004. Currently, she is an Associate Professor of Electrical Engineering at UFMG. Her areas of interest and research include computer networks, telecommunications and computer network management, and wireless sensor networks. She is an expert in telecommunications management network (TMN). Since 1993 she has participated and coordinated research groups on TMN. Currently, she is coordinating the WSN group in the Electrical Engineering Department at UFMG and is heading the MANNA research team.

José Marcos S. Nogueira is an associate professor of Computer Science at UFMG, Brazil. His areas of interest and research include omputer networks, telecommunications and computer network anagement, and software development. He received a B.S. egree in Electrical Engineering, a M.S. degree in Computer Science rom UFMG in 1979, and a Ph.D. degree in Electrical Engineering rom the University of Campinas, Brazil in 1985. He eld a post-doctoral position at the University of British Columbia, anada, 1988-1989, and currently is on sabbatical at the University f Paris VI. He headed the Department of Computer Science at FMG from 1998 to 2000. Currently, he heads the computer network roup at UFMG and was the technical coordinator of the System or the Integration of Supervision (SIS) Project where a complex nd distributed system for the

management of telecommunications etworks was developed. He has served in various roles, including eneral Chair (1985) and TPC Chair (1999 and 2004) of the razilian Symposium on Computer Networks (SBRC), and General hair of LANOMS 2001. He has been a TPC member in IEEE/IFIP OMS (2000, 2002, 2004), IEEE/IFIP IM 2003, IEEE LANOMS 1999, 2001, 2003), IEEE/IFIP MMNS (2000, 2001, 2002, 2003), POM 2002, SBRC (from 1990 to 2004), and IEEE/IFIP DSOM 2003. He is a member of the Brazilian Computer Society (SBC) and Brazilian Telecommunications Society (SBrT). He also participates in the IEEE ComSoc CNOM interest group. He has supervised a number of Ph.D. and Master's students, and publishes regularly in international conferences and journals.

Antonio A.F. Loureiro holds a B.Sc. and a M.Sc. in Computer Science, both from the Federal University of Minas Gerais (UFMG), and a Ph.D. in Computer Science from the University of British Columbia, Canada. Currently he is an Associate Professor of Computer Science at UFMG. He was the General Chair of the Brazilian Symposium on Computer Networks in 2000, Technical Program Chair of LANOMS 2001 (Latin American NOMS) and Brazilian Workshop on Wireless Communication 2002. His main research areas are network management, mobile computing, and distributed algorithms.