Reviewing Dependability Issues and Suitable Solutions for Emerging Wireless Sensor Networks Applications

Carlos Oberdan Rolim[1,3], Edison Pignaton de Freitas[1,2], Tales Heimfarth[4], Carlos Eduardo Pereira[1], Cláudio F. R. Geyer[1] , Armando Morado Ferreira[5], Flávio Rech Wagner[1], Tony Larsson[2]

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

[2]IDE – Halmstad University, Halmstad, Sweden.

[3]Departamento das Engenharias e da Ciência da Computação – Universidade Regional Integrada (URI) – Santo Ângelo, RS – Brazil

[4]Departamento de Ciência da Computação – Universidade Federal de Lavras (UFLA) – Lavras, MG – Brazil

[5]Programa de Pós-Graduação em Engenharia de Defesa – Instituto Militar de Engenharia (IME) – Rio de Janeiro, RJ, Brazil

{carlos.oberdan,epfreitas,flavio,geyer}@inf.ufrgs.br,tales@dcc.uflr.br
cpereira@ece.ufrgs.br, armando@ime.eb.br, tony.larsson@hh.se,

Abstract. *Some Wireless Sensor Networks applications can be considered safe critical due to the important impact caused in occurrence of their malfunctioning. Motivated by this issue, a research area in dependability applied to WSN is gaining strength. As highly distributed systems, WSN present a number of issues that have to be handled in order to achieve a dependable behavior. These issues are stressed considering the current sophisticated WSN, composed by static and/or mobile nodes. This paper proposes a review of the main problems and trends in the area of dependability for WSN, aiming to find suitable solutions to the problems faced by emerging WSN composed of static and mobile sensor nodes. Such applicability is analyzed for an application scenario of area surveillance.*

1. Introduction

Wireless Sensor Networks (WSNs) are being employed in a number of applications nowadays [Kuorilehto et al 2005], which reflects the result of advances in the sensor technologies and wireless communications [Mini et al 2004]. An interesting approach in WSN is the combination of different types of sensors in a single network. The different sensors can provide complementary data which are combined to provide the users with information with more aggregated value, as the example provided in [Heinzelman et al 2004]. Within this context, a new trend that is starting to be explored is the combination of static and mobile sensor nodes [Erman et al 2004]. As an example, a surveillance system can be composed of a number of unattended static ground sensor nodes and mobile sensor nodes carried by Unmanned Ground Vehicles (UGVs) or Unmanned Aerial Vehicles (UAVs) [Erman et al 2004]. A problem in this scenario is how to ensure the correctness of the overall system if some sensors fail. This issue is common to all

WSNs and is being considered a challenge by the research community in the dependability area [Koushanfar et al 2002]. Thus, this is a research area with great potential to be explored, as by now not many proposals address the problem.

Wireless communications per se are already subject to a number of circumstances that leads to failures [Mini et al 2004]. Moreover, sensor nodes typically used in WSNs are very constrained in resources, fact that has its rationale in the attempt to keep both their size and cost as small as possible [Koushanfar et al 2002]. Having these resource constraints in mind, ordinary Triple Modular Redundancy (TMR) based solutions for the sensor hardware are out of consideration. One may argue that if the hardware used in sensor nodes is cheap; providing redundancy of something cheap would still result in something cheap. This is a misleading idea, because the sensor nodes are cheap because they use only the minimum set of components necessary to their operation. Once these components are duplicated or triplicated, the statement that the sensor node is cheap may not hold. Besides the fact that the redundant hardware would require space, compromising the requirement of keep them in a small size. These facts make WSNs very susceptible to faults in the individual nodes, which if not properly handled, may lead to catastrophic consequences depending on the application.

Conversely, besides the mentioned fragilities that WSNs present, which make them susceptible to failures, they also count with a feature that makes them able to be robust and fault tolerant. This characteristic is related to the great number of nodes used in WSN deployments. This fact provides support to redundancy, which can be used to overcome faults in the sensor nodes that lead to errors in the communications which finally can manifest a system failure in the final service that the network delivers.

Still important to consider in this context is the fact that many WSNs are autonomous systems. This fact makes part of both the problem and the solution in terms of dependability, at the same time. On one hand, as autonomous (complex) systems, equipped with decision making mechanisms, they are able to operate without human intervention. However, a concern about dependability is raised exactly because of these decision mechanisms, which may contain faults that trigger the rest of the error-failure-fault-error… chain compromising the system functionality [Avizienis et al 2001]. On the other hand, autonomous WSNs are able to provide self-healing features and thus recover from failures of individual nodes, handled as faults from the system perspective. This allows the system deliver correct functionality, in spite of presence of faulty nodes.

This paper proposes a review of the main problems faced in the operation of WSN problems that may compromise the system availability and reliability. These two components of the dependability concept are considered the primary ones for WSNs, as stated in [Taherkordi et al 2006], and represent a great potential threat to emerging WSN applications. Additionally, it presents prominent approaches to address some of these related issues and an analysis of their applicability to a surveillance system. Section 2 presents the surveillance system considered in the present study. In Section 3 an overview of the considered dependability problems in WSN is presented. Section 4 provides a small survey of outstanding approaches that propose solutions to handle these problems, focusing on the fault detection aspect, while Section 5 discusses the applicability of a selected approach to the presented system. In Section 6 the conclusions are drawn and the future works are outlined.

2. Application Scenario: Area Surveillance System

A surveillance system considered in this study is composed of two main types of sensors: static on ground and mobile in the air (UAV-carried, from now on just called "UAV"). There are G static ground sensor nodes spread on the area of interest, which are individualized by an identifier sn_i, (i = 1,…, G). The ground static sensor nodes are spread according to a given distribution, which can be random or uniform following a defined pattern. N UAVs fly over the area, following a random or predefined movement pattern. These UAVs are identified by u_i (i = 1,…, N). G, the number of ground sensor nodes, is assumed to be much bigger than the total number of UAVs, i.e. N. The two groups of sensor nodes are further divided according to their sensing capabilities. The considered capabilities are the type of sensing measurements that they can perform and the sensing range. For the ground sensor nodes, the considered measurements are, for instance, vibration, temperature, humidity, difference in the magnetic field, and acoustic signature. Examples for the UAV-carried sensors are visible light cameras, infrared cameras, and SAR/ISAR radars. The sensing range is a tunable parameter that remains constant for the ground sensor nodes, but may vary for the UAV-carried ones, according to the sensing device. The sensor nodes wirelessly communicate with each other within a given tunable range, called communication range. Due to the broadcast nature of wireless communications, all nodes inside the range of another receive a message issued by the sender node.

The system works as follows: The ground sensor nodes are configured to detect possible targets, which is defined by a set of threshold levels of its measurements. When the acquired measurements reach a configured threshold level, a "match" with the detecting criteria is achieved. In the occurrence of a match, the sensor node issues an alarm, which is received by all nodes in its vicinity that are within its communication range.

Alarms are represented by single communication packets. These packets contain a timestamp, the position of the issuer node, and the type of the possible target. The two first components of the alarm allow its unique identification, avoiding alarm duplications. For the purposes of this work, each alarm indicates one target. This means that if it is a group of persons or vehicles, they are handled as a unique entity.

The main elements of the scenario and the described system are presented in Fig. 1. In the figure it is possible to observe the occurrence of a detection of a possible target by a ground sensor node. This node issues an alarm that is received by all other nodes in its communication range. One of these neighbors relays the alarm which is received by a UAV, characterizing the alarm delivery.

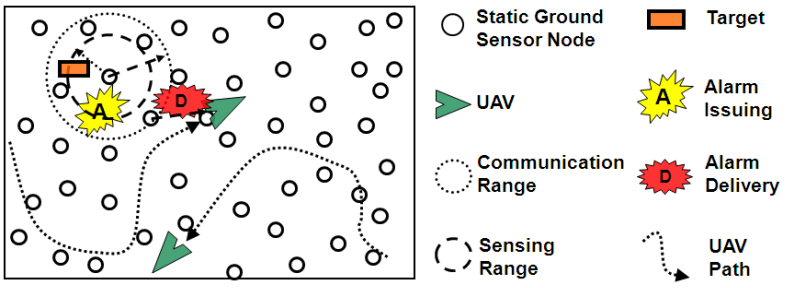

Figure 1. Overview of the application scenario.

By the occurrence of an alarm, the goal of the system is to allocate one of the UAVs, which are equipped with more sophisticated sensors, to fly towards the area where the alarm was issued, gather further information about the possible target, and confirm it as a target, i.e. an intruder or a threat.

Due to an uncontrolled or even harsh or hostile environment, static on ground sensors can become faulty and unreliable. A faulty sensor node can propagate erroneous collected data and affect the correctness of overall system. Another possible problem is that faulty nodes may affect the alarm forwarding. Therefore, the surveillance system needs to perform the identification of fault nodes. This is not a trivial task and encompasses the selection of a suitable algorithm that can be applied in the current scenario.

In next section the background information for the dependability problem in WSN is presented to support the selection of the algorithm further discussed.

3. Dependability Issues in WSN

The notion of dependability is broken down into six elementary component properties [Avizienis et al 2001] as follows: a) Availability – the system will be operational when needed; b) Reliability – the system will keep operating correctly while being used; c) Safety – the system operation will not be dangerous; d) Confidentiality – there will not be disclosure information when not authorized; e) Integrity – there will be no unauthorized modification of the information used by the system; and f) Maintainability – the capability to perform a successful repair action within a given time.

Talking about WSN, the two first aspects of dependability are a must [Taherkordi et al 2006]. It does not mean that the other ones are not important, but these two are the ones that concentrate the focus of attention of the research community, as shown in the literature [Taherkordi et al 2006]. Some of the others aspects correspond to complete different research areas, which is what happens for instance with confidentiality. On the other hand, maintainability may be also considered as a "hot topic", as there are several proposals that aim at reprogram the entire network, which can be seen as a way to fix the software pre-installed in the sensor nodes, such as [Fok et al 2005] [Liu et al 2003]. However, they do not talk about dependability issues in their work, so it is possible to say that this capability is a "side-effect" of such proposals, even its application with this goal is definitely reasonable and valid.

Most of WSN applications should run continuously and correctly during their operation. They are characterized by being mission critical, complexly structured, composed of components that have significantly fault rates, such as semiconductor integrated circuits-based systems and sensor devices, besides their software part represented by the decisional mechanisms [Koushanfar et al 2002]. All these together reveal a big threat to WSN operation, thus compromising the system availability and reliability, supporting the strong interest by and focus on these two topics.

A WSN-based system is basically composed of four elements: 1) the individual sensor nodes; 2) the network formed by these nodes; 3) the sink node(s); and 4) the backend system. The first two elements are the essential parts of a WSN, while the third is the way the data collected by the sensor nodes can be accessed from the outside of the WSN, and finally the fourth is the information system that presents the data to the final user. In this work the considered problems are only the faults that occur in the first two.

Problems related to the sink(s) and the information systems are out of the scope, even considering that they are important parts of the system.

3.1. Faults

Within the delimited scope, the source of faults in WSN can be then classified as: 1) Node Faults and 2) Network Faults.

3.1.1. Node Faults: Node faults are considered to be internal faults that occur in individual sensor nodes. These faults may have their source in the hardware or in the software component of the nodes. In the hardware, common problems are related to antennas that are fragile and cannot resist to impacts [Langendoen et al 2006] and to batteries that provides incorrect voltage or current outputs damaging other internal components [Tolle et al 2005].

In spite of hardware faults representing an important problem, software ones are very common and may even exceed the first ones, depending on the complexity of the WSN application. As deeply embedded systems, sensor nodes are hard to program due to the inherent resource constraints of such systems. Incautious programming of sensor nodes may easily lead to memory overruns, buffer overflows, deadlocks and to all sorts of trick and hard to debug programming problems. Incorrect handling of data provided by the sensor unit can be caused by faults in the application programs. These faults in one node may trigger a fault in data aggregation software in other nodes, as the WSN in general run distributed applications.

3.1.2. Network Faults: Network faults are those related to the interaction among nodes and that lead to an error that affects the nodes' communication and coordination. One of the very common sources of fault is radio interference. This problem may occur by obstacles present in the environment, or by the presence of other devices operating in similar frequencies, or even by other nodes of the network displaced close by. In fact this last case, even a legitimate emission by a neighbor node may cause undesired interferences when frequent collisions of messages happen. Link stability is also reported as a problem in sensor nodes networking. This instability affects the sensor nodes routing capabilities, which per se is another source of problems.

Routing issues can also be considered as a significant source of networking malfunctioning. In WSN, routing has a great importance, specially related to the semantics of the applications that run over the networks. Faults in the routing protocols may compromise the network as a whole, or parts of it, in a number of ways (e.g. the wrong cluster-head election in hierarchically organized WSN due to delayed, dropped or misrouted messages).

It is important to observe that this classification complies with the fundamental concepts in dependability presented in [Avizienis et al 2001]. For example, the software faults discussed above, such as buffer overflow, can be classified as a fault due: Development; Internal; Human-made; Software; Non-Malicious. It can be classified as presented because it is an internal part of the system, a software part, programmed by a person that did not consider checking the input value, so it was human-made introduced, and not necessarily malicious. It could be malicious if the programmer intentionally considered the hypothesis of such condition happens, nevertheless did not introduce the value checking. However, this work focuses on the sensor network perspective. This means that the categorization of the faults tries to emphasize the elements present in

these systems, and it is why it does not mention much the classification presented in [Avizienis et al 2001].

3.2. Errors

Errors are deviations of the normal way that a part of a system should work, and they are activated by faults. The types of considered errors in this work follow the consequences of the faults presented above, as they are the ultimate responsible for the occurrence of the errors.

As an example of an error caused by a node fault, it can be mentioned the miscalculation of a value based on wrong data provided by the sensor device. The sensor device may provide an unexpected value, which is out of the range of tractable values for the piece of software that manipulates this data. As a result, an erroneous output is generated being a manifestation of two original faults: the former is a hardware fault, related to the sensor device that provided the wrong value; and the latter is the fault due to lack of protection in the program code against out of bounds values, which leads to an erroneous state.

Errors due to network faults can be exemplified by selection of wrong routes to forward messages due to routing faults. The consequence of these faults can be erroneous data aggregation, for example, or undelivery of expected messages. Malformed or incomplete communication packets can be presented as examples of errors due to interferences.

3.3. Failures

A failure represents an incorrect delivery of a service provided by a system. Originally motivated by one or more faults, it is the external manifestation of an error.

In WSN a failure could be considered the final erroneous information provided to the end user, which is ultimately the desired service from a WSN. However, for the scope of this work, the considered failures are related to the nodes and the network, as discussed above and classified according to [Souza et al 2007]:

Crash or Omission Failures: These are failures in which the nodes or parts of the network stop to respond to requests, e.g. due to the exhaustion of nodes' batteries.

Timing Failures: They are failures that occur due to mismatch between the timing requirements specified for the WSN services and the actual observed timing behavior.

Value or Semantic Failures: These types of failures represent deviations in the expected outputs of the system in terms of the content of data or information.

Arbitrary or Byzantine Failures: These are types of failures that do not follow a consistent pattern, such as an intermittently correct and erroneous sensor output.

4. Prominent Approaches to Handle Dependability in WSN

The study about dependability in WSN is gaining importance as the applicability of this technology is increasing, but there is still a gap between problems and solutions when it comes to assure dependability in WSN operation.

This section presents some selected prominent approaches to deal with dependability problems in WSN. The bibliography review presented here focuses on the

fault detection and forecast techniques. Fault recovery and removal are for sure important, but the goal of the paper is kept in the mentioned subject. The idea is analyze to suitable fault detection and/or forecast techniques to be used in emerging applications of WSN. It does not mean that other existing techniques, especially for fault recovery, could not be considered to be (re)used. However, they are kept as related works instead.

The works described in this section are classified following the taxonomy presented in [Souza et al 2007].

Automatic fault detection and forecast techniques have the goal to provide a system with the ability to assess if its functionalities are correct and if it will continue working in the future. Some proposals provide both techniques together.

Following a similar reasoning which based the classification of faults according to node or network ones, the analysis of the fault detection techniques presented here considers also the elements involved in this process. This is related to the considered granularity. Faults detected by the nodes themselves are classified as self-diagnosis. Fault detection performed by the interaction of nodes is classified in two classes: group detection and hierarchical detection. The former consists of mechanism in which a number of nodes are used to monitor other node(s). The latter considers the usage of a hierarchical detection tree, among other mechanisms, which shifts the performance of the detection to more powerful nodes outside the WSN, for instance a sink. Group detection techniques are especially interesting for the type of WSN target of this work, so more attention will be given to this type of technique.

4.1. Self-Diagnosis

Self-diagnosis techniques consider the use of the nodes themselves to detect and in many proposals forecast the occurrence of a fault. These techniques consist of performance of tests in the system elements so that the faults can be detected. In [Hart et al 2005] tests using accelerometers are used to detect possible faults due to impacts.

An important source of node failure is energy exhaustion. Keeping track of the energy resources may indicate the faults in the nodes functionalities. In [Rakhmatov et al 2001], an estimation of the nodes' batteries lifetime is performed. This approach uses the measurement of the current battery voltage, analyzing the discharge curve and the current rate to determine estimation for the battery exhaustion.

In [Chen et al 2006] nodes are able to detect faults in their sensor devices and inform other nodes in the network that they are not able to perform sensing services anymore. However, as their transceivers and batteries are still working, they are able to perform routing, and provide forwarding of incoming packets.

4.2. Group Detection

A technique to detect faulty sensors in a network of mobile sensors used for target tracking is presented in [Kim et al 2008]. The technique is used to identify faults in sensors readings when estimating the targets positions. In order to detect the nodes that are providing an incorrect target position, two algorithms run concurrently: the former is a semi-decentralized dynamic data fusion, which employs a median-consensus using non-faulty nodes to provide the estimate target position, while the latter one, is a fault detection algorithm, which uses the estimation provided by the former to decide if a given node should be considered as faulty or not.

Clustering is an important mechanism largely used in WSN. In [Taherkordi et al 2006] a mechanism to detect faulty nodes in clustered WSN is proposed. This approach uses an event-based middleware software architecture for the services provided by the nodes, and distinguishes those services according to the hole of a node in the network, i.e. cluster-head or cluster member. The former are considered event-brokers while the latter are event-sources. By using a publish-subscribe scheme for the nodes' interaction, the proposal is to detect node disconnection faults due to node mobility or energy depletion. Additional mechanisms make possible fault recovery during reconnections.

Exploring the fault detection abilities of grouped nodes, [Chen et al 2006] proposes an approach in which the nodes in agiven area test each other to detect the occurrence of faults. According to the specific deployments, the nodes generate test results for all their neighbors. Based on this information, they decide about their own tendency. Then a consistency check is performed to decide about the obtained results. This proposal allows the detection of faults in the sensing measurements, which generate erroneous values, but also faults that lead to crash failure.

ASFALT (A Simple Fault-tolerant signature-based Localization Technique for emerging sensor networks) [Jadliwala et al 2007] proposes a fault-tolerant location mechanism for sensor networks. The work as whole deals with problems from nodes' deployment to the reduction of the error in signature-based location algorithms. The part related to the fault detection is performed in a group fashion, in which a group head keeps the information about the health status of the nodes in its group. A group with healthy nodes is capable to deliver correct values to solve the location of a target. Group heads exchange their status among them via a periodic broadcast. A fault is detected when less than three groups are declared healthy to solve the location of a given target.

In [Ni et al 2009] a Bayesian model to detect erroneous data provided by faulty sensor nodes is presented. The approach is divided in two phases, in which the former uses a Bayesian MAP criterion to determine a set of aggregation node subsets, which will perform the decision in the latter whether a given data is erroneous or not. Suspicious data is tagged as so, and by passing through the aggregating nodes, the audition of its content is done. As the proposal is highly computing intensive, the authors relaxed the timing requirements providing a semi-realtime approach, in which the data integrity audit is performed much less frequently than the sensor samples are taken.

Neural networks are used to identify the misbehavior of sensor nodes in [Obst 2009]. The authors propose a neural network training based on spatially organized distributed computation in which a local interaction among a group of nodes is explored. The training is performed in two steps: 1) a sample matrix of internal network states and the corresponding output matrix with activations are used; and 2) the output weights are computed. The fault detection is done by the comparison of the predicted values provided by the neural network and the actual readings. If the difference between them exceeds a threshold, a fault it detected.

4.3. Hierarchical Detection

A fault detection mechanism using management architecture called MANNA is proposed in [Ruiz et al 2004]. This work uses a manager located externally to the WSN, which has the global view of the entire network. This vision allows the execution of

complex management tasks. Every change in the nodes status is informed to the manager, which is responsible to test the sensor nodes by sending commands to them. As drawback, besides the centralized manager, this approach has a great overhead due to the number of communications among the sensor nodes and the manager.

In [Ringwald et al 2006] a debugging tool that uses a decision binary tree is proposed. Verifications in the tree are performed so that possible faults can be detected. The authors claim that the one of the major pros of their approach is the fact that no additional traffic has to be transported among the sensor nodes to provide the detection.

A detection tree in which the child nodes report to their parents and these continuously forward up to the sink is presented in [Rost et al 2006]. When a node receives the status of its child nodes, it performs an aggregation with its own status and forwards the result. This mechanism continues until the information arrives to the sink.

In [Chitnis et al 2009], a study proposing the use of some powerful nodes together with resource constrained ones in a WSN is analyzed. This proposal uses such a heterogeneous composition of the WSN to improve reliability of data aggregation, by the use of the powerful nodes to detect and recover from faults. The authors compare a detection aggregation tree structure to a hybrid one in terms of cost-benefits and performance, regarding faults. Their conclusions show benefits in using a number of expensive powerful nodes to achieve the desirable behavior against faults.

5. Case Study – Analysis of Suitable Approaches

This section presents the used fault model and the algorithm that has these characteristics and can be used in the proposed problem-scenario presented above.

5.1. Network and Fault Model

The model used in this work considers a scenario in which the ground sensor nodes are deployed in a rectangular area where they are randomly distributed.

The instance of the model studied in this paper considers an area of 10 Km x 10 Km, and sensor nodes with 350 meters communication range. For these parameters, the minimum number of nodes that provide a connected network is 5000. This is a requirement to guarantee the correct system functionality, as discussed in [Freitas et al 2010]. This number is obtained by using:

$$P(d_{min} > 0) = \left(1 - e^{-\rho\pi r^2}\right)^n \quad (1)$$

where $P(d_{\min} > 0)$ is the probability that the minimum degree of the network is higher than zero, "ρ" is the node density, "r" is the communication range, and "n" is the total number of nodes in the network.

The formula presented in (1) determines the probability that a network of nodes randomly deployed with independent uniform probability (homogeneous Poisson point process in two dimensions, which generates a geometrical random graph) has a degree greater than zero, which means that each node in the network has at least one node via which it may communicate with any other node in the network [Bettstetter 2002]. Thus each node can locate neighbors within its transmission range to infer about faulty ones.

To attend an alarm occurrence the system can allocate one of six UAVs of three different types, equally distributed, that patrol the area, flying at altitudes not higher

than 250 meters and with speeds from 100 Km/h up to 120 Km/h. The UAVs have a communication range of 1.5 Km.

This work considers two different types of failures for the ground static sensor nodes: crash failures and value failures. The former considers that the node is permanently enable to perform its activities which can be detected, for instance, if it does not respond any request for more than a certain time period. The latter considers that a node is providing erroneous sensor readings and thus is not trustable to detect a possible event of interest and issue a corresponding alarm. For this second type of considered failure, the node is still capable to route messages, so incoming messages with alarms from other nodes can still be forwarded via this node.

For each of the considered failures it is established a possible situation. A number of such failures occur in a given region of the network, so that it creates an island of faulty nodes. For the first type of failure, it results in incomplete absence of the network in that region, as the nodes cannot even forward alarms. For the second one, the network still exists, but the sensing ability is out of order, so the system functionality is broken.

The goal is to use a fault detection mechanism that is able to provide the information about the occurrence of such situations, so that fault recovery measures can be applied. This is not a goal of this paper to deeply analyze such recovery measures, but as an example of a possible solution can be the use of the mobile sensors, the UAVs, to cover the problematic areas. The nodes that detected such problems can issue an alarm calling a UAV to fly close by the faulty region in order to cover the region, or at least make the UAV have that region as a preferred flying zone. It is important to highlight that the aim is to preserve the system functionality in providing the final user information about possible targets. Thus, the use of the UAVs to cover the areas with faulty nodes is a reasonable solution. However, this paper just focuses on fault detection on ground static sensors and this mentioned possible solution was given as an example.

5.2. Applied Fault Group Detection

After presenting prominent approaches in dependability research area for WSN and the considered model for this research work, the proposal is to use group based fault detection to handle the problems of the scenario depicted above in Section 2. This choice is due to the large amount of static sensor nodes on the area where the system is deployed, in which each node is able to locate neighbors within its transmission range and use its results to infer about faulty neighbor nodes.

The scenario also requires that the fault detection needs to be conducted in a real-time mode, with low latency, high throughput and, due to the limited energy resources of the sensor nodes, the detection of faulty nodes cannot be expensive in terms of energy. Such restriction excludes centralized approaches in which a base station is used to collect information from all sensors and calls for a distributed and more generic algorithm that explores local decision taken by the sensor nodes. A suitable solution with the desired distributed characteristics is an algorithm presented in [Chen et al 2006].

The proposed algorithm is depicted is Figure 2. It uses five steps to infer about status of grouped sensors. In step one, every sensor S_i tests the set of its neighbors $N(S_i)$. This test is done using measurements of each sensor´s neighbor with two

variables, d_{ij} and Δd_{ij}, and two predefined threshold value $\theta 1$ and $\theta 2$. If the modulus of measurement difference between S_i and S_j at a time t is greater than threshold $\theta 1$ then the historical data is used to find if the current measurement changes over the time significantly from previous measurement. In affirmative case, it is more likely the sensor is faulty. If this variation of measurement is greater than threshold $\theta 2$ the test results in 1 (indicating that S_i and S_j are most likely in different status) otherwise in 0 (indicating that either both S_i and S_j are most likely good or faulty). In step two S_i generates a tendency value T_i based upon its neighboring sensors' test value. If the sum of test results is less than half of the number of its neighbors, the sensor is likely good (LG). Otherwise, it's likely faulty (LF). To finish this step T_i is communicated to neighbors. In step three, the number of the LG sensors with coincident test results determines whether the sensors are good (GD) or faulty (FT). If the summation of LG sensors is greater or equal than half of the number of its neighbors then T_i is set to GD and so T_i is communicated to neighbors. In step four GD sensors are used to determine status of the remaining undetermined sensors. Finally, step five does a validation check to make sure the diagnosis is consistent throughout the entire network. The next subsection presents the analysis of the application of this algorithm to detect faults in the surveillance system presented in Section 2.

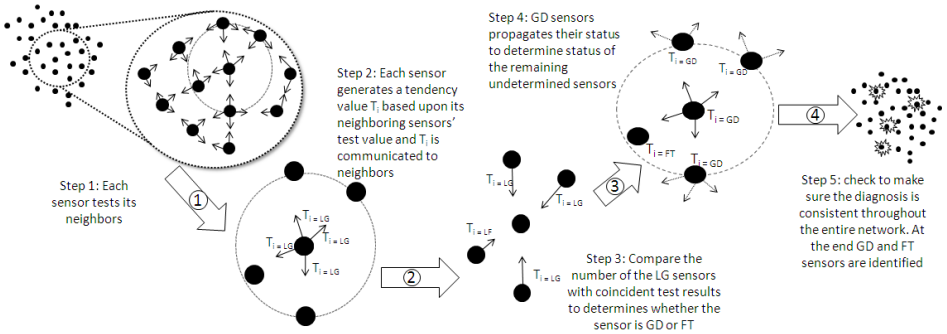

Figure 2. Overview of the Fault Detection Algorithm.

5.3. Analytical Evaluation

Based on the results presented by the authors of the technique in [Chen et al 2006], and on the results about the proposed surveillance system degradation by a non-sufficient number of ground sensor nodes according to (1) [Freitas et al 2010], it is possible to analyze the applicability of the proposed fault group detection to this particular system.

The results presented in [Freitas et al 2010] analyze an instance of the network with the same parameters as described above. The achieved results showed that with a number below 50% of the ideal number of nodes, the system performance becomes really degraded, approaching values below 60%, and dropping fast from this point to lower number of nodes. For a number of nodes above 50% of the ideal amount, the system responds with values above 70% of efficiency, which can be considered a fairly good result. Table 1 presents this data. This data emphasizes the problem in losing sensor nodes, especially from the level of 50% of the ideal number of nodes. Assuming the first type of the considered failures, the node crash, a detection of a faulty region should be triggered when a group of nodes detects in their vicinity a number of approximately 50% of faulty nodes. For the parameters presented in the example instance of the considered system, each node has in average 18 neighbors, which is a

value deduced from the average number of nodes in a region within an area of the circle determined by the communication range of a sensor node, calculated by:

$$n = \left({}^{N}/_{A}\right) \cdot \pi r_c^2 \quad (2)$$

Where n is the average number of nodes in a region with area equivalent to the circle determined by the communication range of a sensor node, N is the total number of nodes distributed in the area, A is the considered area, and $r_c$is the communication range of sensor node. From the achieved value, it is subtracted one unit, which is the reference node in the center of the circle and the result is the number of its neighbors.

Table 1. Relation Among Number of Nodes, P(d_{min} > 0) and System Efficiency

Number of Nodes	% of Nodes in Relation to the Ideal Number	$P(d_{min} > 0)$	Normalized System Efficiency in Average
5000	100%	1.0	85%
4500	90%	0.999863268	85%
4000	80%	0.999168572	85%
3500	70%	0.995031912	83%
3000	60%	0.971209180	76%
2500	50%	0.846539727	65%
2250	45%	0.675534269	53%
2100	42%	0.521033789	39%
2000	40%	0.401629927	34%
1500	30%	0.009203929	<10%

Considering the area covered by the neighbors of these neighbors, the number of nodes may come to around 70. This number may vary depending on the nodes' distribution, but in average this is a good approximation for the simulation parameters mentioned above. Considering a sensing range of 100 meters, which is a realistic value considering available sensor technology nowadays [Wilson 2005], this amount of nodes covers a representative part of the total area, and as so, a failure in covering it, may result in a degraded system functionality. For this concrete case, applying the approach presented in Section 5.2, which was proposed in [Chen et al 2006], if a number of faulty nodes above 35 is detected in a given region (50% of the nodes of the considered area), a fault can be notified. This threshold was established based on the fact that when a percentage below 50% of the total number of nodes in the network the functionality is seriously degraded, the same holds for a part of the network.

Considering de second type of failure, the same reasoning described above can be applied. Even if the nodes are still able to communicate; they are only able to forward alarms from other regions. This means that the region where they are located will become without sensing coverage. Thus, even that the alarm delivery mechanism is not affected, the detection of possible targets is, which indeed represents a malfunctioning of the system.

6. Conclusions and Future Work

This paper proposes a review of the efforts in handling dependable issues in WSN, and their applicability in emerging WSN applications. A review of important concepts was presented, along with a survey of prominent works in the area. Finally a highlight on

one of them was done, which was deeply described and had its applicability analyzed to an application scenario of a surveillance system.

As future works, other mechanisms have to be combined in order to address the different types of nodes that compose the network. In the presented application, this difference was the nodes' mobility. A second issue is the need for a more accurate metrics to determine the dependability qualities of the proposed solutions. Finally, more than combined solutions, unified solutions that consider different types of sensors need to come up to address these types of emerging applications.

References

Kuorilehto, M., Hännikäinen, M. and Hämäläinen, T.D. (2005) "A Survey of Application Distribution in Wireless Sensor Networks", EURASIP Journal on Wireless Communications and Networking, 38(5), p.774-788.

Mini, R., Loureiro, A., Nath, B. (2004) "The Best Energy Map of a Wireless Sensor Network", In: Proceedings of the 7th ACM Intl Symposium on Modeling, Analysis and Simulation of Wireless and Mobile System, Venice, Italy, p.165-169.

Heinzelman, W. B., Murphy, A. L., Carvalho, H. S. and Perillo, M. A. (2004) "Middleware to support sensor network applications, Network", IEEE, vol. 18, no. 1, p.6-14.

Erman, A.T., Hoesel, L. and Havinga P. (2008) "Enabling Mobility in Heterogeneous Wireless Sensor Networks Cooperating with UAVs for Mission-Critical Management", IEEE Wireless Communications, vol. 15, is. 6, p.38-46.

Koushanfar, F., Potkonjak, M. and Sangiovanni-Vincentelli, A. (2002) "Fault Tolerance in Wireless Ad-Hoc Sensor Networks", IEEE Sensors, vol. 2, p.1491-1496.

Avizienis, A., Laprie, J. and Randell, B. (2001) "Fundamental Concepts of Computer System Dependability", IARP/IEEE-RAS Workshop on Robot Dependability: Technological Challenge of Dependable Robots in Human Environments, Korea.

Taherkordi, A., Alkaee Taleghan, M. and Sharifi, M. (2006) "Dependability Considerations in Wireless Sensor Networks Applications", In: Journal of Networks, v.1, n. 6, Academy Publisher.

Fok, C.L., Roman G.C. and Lu, C. (2005) "Rapid Development and Flexible Deployment of Adaptive Wireless Sensor Network Applications", In: Proceedings of the 24th International Conference on Distributed Computing Systems, p.653-662.

Liu, T. and Martonosi, M. (2003) "Impala: A Middleware System for Managing Autonomic, Parallel Sensor Systems", In: Proceedings of the 9th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, p.107-118.

Langendoen, K., Baggio, A. and Visser, O. (2006) "Murphy loves potatoes: experiences from a pilot sensor network deployment in precision agriculture", In IPDPS 20th International Parallel and Distributed Processing Symposium.

Tolle, G., Polastre, J., Szewczyk, R., Culler, D., Turner, N., Tu, K., Burgess, S., Dawson, T., Buonadonna, P., Gay D., and Hong, W. A. (2005) "Macroscope in the Redwoods", In: SenSys '05: Proceedings of the 3rd international conference on embedded networked sensor systems, New York, NY, USA, p. 51–63.

Souza, L. M. S., Vogt, H. and Beigl M. (2007) "A Survey on Fault Tolerance in WSN", Interner Bericht. Fakultät für Informatik, Universität Karlsruhe.

Harte, S. and Rahman A. (2005) “Fault Tolerance in Sensor Networks Using Self-Diagnosing Sensor Nodes”, In: IEEE International Workshop on Intelligent Environment, p.7–12.

Rakhmatov D. and Vrudhula S. B. (2001) “Time-to-Failure Estimation for Batteries in Portable Electronic Systems”, In: Proceedings of the 2001 International symposium on Low power electronics and design, p.88-91.

Chen, J., Kher, S. and Somani A., (2006) Distributed fault detection of wireless sensor networks. In Proceedings of the 2006 workshop on Dependability issues in wireless ad hoc networks and sensor networks, p.65-72.

Kim, Y., Gu, D., and Postlethwaite, I. (2008) “Fault-Tolerant Cooperative Target Tracking in Distributed UAV Networks”, In: Proceedings of 17 IFAC conference, p.8878-8883.

Jadliwala, M., Upadhyaya S. and Taneja, M. (2007) “ASFALT: A Simple Fault-Tolerant Signature-based Localization Technique for Emergency Sensor Networks”, In: Proceedings of 26th IEEE International Symposium on Reliable Distributed Systems, p.3-12.

Ni, K.S., Pottie, G.J. (2009) “Sensor Network Data Fault Detection Using Bayesian Maximum a Posterior Sensor Selection and Hierarchical Bayesian Space-Time Models”, Technical Reports, Center for Embedded Network Sensing, UCLA.

Obst, O. (2009) “Distributed Fault Detection using a Recurrent Neural Network”, In: Proceedings of the 2009 International Conference on Information Processing in Sensor Networks (ISPN'09), San Francisco, CA, USA, p. 373-374.

Ruiz, L. B., Siqueira, I. G., Oliveira, L. B., Wong, H. C., Nogueira, J. M. S. and Loureiro, A. A. F., (2004) “Fault management in event-driven wireless sensor networks”, In: Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems, p.149-156.

Ringwald, M., Römer, K. and Vitaletti, A. (2006) “Snif: Sensor network inspection framework”, Technical Report 535, ETH Zurich, Institute for Pervasive Computing.

Rost, S. and Balakrishnan, H. Memento. (2006) “A health monitoring system for wireless sensor networks”, In: Proceedings of SECON.

Chitnis, L. Dobra, A. and Ranka S. (2009) “Fault tolerant aggregation in heterogeneous sensor networks”, In: Journal of Parallel and Distributed Computing, Vol. 69, Is. 2, p. 210-219.

Freitas, E.P., Heimfarth, T., Pereira, C.E., Ferreira, A.M., Wagner, F.R. and Larsson, T. (2010) “Experimental Analysis of Coordination Strategies to Support Wireless Sensor Networks Composed by Static Ground Sensors and UAV-carried Sensors”, In: Proceedings of IEEE International Symposium on Parallel and Distributed Processing with Applications. Taipei, Taiwan, p. 152-161.

Bettstetter, C. (2002) “On the Minimum Node Degree and Connectivity of a Wireless Multihop Network”, In: Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking & Computing, ACM, New York, NY, USA, p. 80-91.

Wilson, J. S. (2005), Sensor technology handbook, Vol 1, Elsevier.