2009 International Conference on Computational Science and Engineering

Minesweeper for Sensor Networks - Making Event Detection in Sensor Networks Dependable

Karima B. Hein, Reinhold Weiss
Institute for Technical Informatics
Graz University of Technology
Graz, Austria
Email: {Hein,RWeiss}@TUGraz.at

***Abstract*—Event detection using Wireless Sensor Networks (WSNs) has become a new field of research in the past years, increasing the need for dependability and fault tolerance. Our work exploits the massive redundancy of large WSNs in combination with neighbours' relations to identify faulty nodes. We present a new approach to categorize nodes in being faulty or fault free based on the event detection results of the nodes' neighbours and the nodes adjacent to the neighbours. For error probabilities $<$ 0.2 our algorithm performs closely to other work in the field, and performs considerably better for error probabilities up to 0.5.**

***Keywords*-Dependability, sensor networks, fault tolerance**

I. Introduction

Sensor Networks are a widespread technology that has gained considerable importance in recent years. Particularly, *Wireless* Sensor Networks (WSN) are used for ensuring the safety of human beings i.e. patient monitoring or structural health monitoring [1]. A WSN is a network consisting of spatially distributed devices, so-called nodes. These nodes are equipped with a number of sensors to cooperatively monitor environmental conditions. The nodes forward their measurements as raw or preprocessed data to a base station. As the user's interface to the WSN, a base station is a device much better equipped with energy and computational resources than a regular node.

Environmental monitoring with WSNs encompasses a vast area of applications, with the most seminal project being habitat monitoring on Great Duck Island [4]. Related applications concerning event detection include forest fire detection [2] and monitoring of volcanic sites [6].

Faulty nodes can propagate incorrect values, misleading neighbouring nodes. In a WSN concerned with event detection, the application running on the base station is often only interested in the presence of events of a certain size. A small number of malfunctioning nodes could convince their neighbours of an incorrect spread of the event, consequently preventing a positive report to the base station.

Especially with WSNs deployed outdoors there are countless possibilities for the introduction of faults starting from incorrect deployment to transient errors caused by dew or shadow. Hence, it is clearly desirable to have more dependable, and thus more robust WSNs.

The service of the WSNs we consider is to detect event regions. Accordingly, a service failure of the system corresponds to the reporting of false positives or false negatives. The algorithm we propose is concerned with diagnosis, paving the way for fault tolerance as well as fault removal. The benefit of fault tolerance is that in spite of the presence of faulty nodes, the network can continue to work correctly. Fault removal can be realised by either flagging or rebooting faulty nodes.

There are two main ways to provide fault tolerance in WSNs: Either exclusively considering the single node or considering the network in connection with the base station. Fault tolerance conducted by the base station implies lots of bidirectional communication, i.e. the base station has to gather many data packets from a large number of

978-0-7695-3823-5/09 $26.00 © 2009 IEEE
388
DOI 10.1109/CSE.2009.146

nodes (usually all nodes) to be able to arrive at an informed decision. Contrarily, considering solely single nodes leads to an autonomous decision involving less overhead.

The core idea of this work is to provide an algorithm for autonomously answering the question 'Is the node faulty or non-faulty?'. Our node-centred system considers the node's own findings, the findings of the node's neighbours and also the findings of the nodes adjacent to the neighbours. The title of this paper was inspired by the Minesweeper game. Our simulator shows a digit at every node (cf. Figure 1 on page 6) indicating the number of neighbours that detect an event, in this manner bearing a resemblance to Minesweeper.

II. Related Work

One of the first publications tackling fault tolerance in event detection is [3], where a distributed Bayesian algorithm is introduced. The authors argue that "measurements due to faulty equipment are likely to be uncorrelated, while environmental conditions are spatially correlated". The authors fail to mention that measurements due to faulty equipment can be correlated with environmental conditions. Consider a cluster of nodes deployed in the field that is shaded by a hedge or a mould: morning dew will stay longer on the shadowed nodes than on their sunbathed neighbours. This is an example where environmental conditions can cause equipment to become (transiently) faulty. In addition, the authors concentrate their considerations exclusively on nodes that lie in the inner region of the event, whereas our work is concerned with nodes situated at the boundary of the region as well.

In [5], Ould-Ahmed-Vall et al. further develop the ideas of [3] by introducing different failure probability levels for each node. Two different error models are considered: one takes into consideration that "nodes that are closer to each other have a higher spatial correlation than nodes that are farther apart"' and the other model considers how the neighbours are geographically distributed around the node. The authors don't comment on any particular circumstances concerning nodes on the boundary. In contrast to [5], our scheme does not only take the node's neighbours' decisions into account, but also the decisions of the nodes adjacent to the neighbours.

This idea of "neighbours' neighbours" has also been considered in [7], where Xiang et al. introduce a distributed weighting scheme for the detection of event regions. The proposed algorithm weights the node's neighbour's findings and their neighbours' findings with two different weights depending on the detection outcome and the distance to the node in question. Complementary to simple weighting, our approach considers weighted proportions of distinctive characteristics of the node and its eight neighbours. Nevertheless, as the idea described in [7] is closest to our approach, we compare our results to those of [7] in Section IV.

III. Minesweeping for Fault Diagnosis

Only the correct identification of a fault enables mitigation of the fault's impact. Consequently, the crucial first step in dealing with faulty nodes is diagnosis.

A. Problem Formulation

We consider a large WSN with the nodes installed in a grid topology. The nodes perform binary event detection, meaning a node only determines if an event is present or not. The measurements taken by a node can be incorrect, resulting in a false positive if an event is detected although there is no event present. Analogous, an event that goes undetected is called a false negative. The aim of this work is to explore if the event detection of each individual node is correct. This is achieved by comparing its decision with the decisions of the nodes in the closer vicinity, i.e. the two-hop-neighbourhood.

B. Assumptions and Context

We only consider binary event detection, that is if the sensor reading is above a threshold, the sensing application's output is '1', i.e. the node detects the event. Otherwise, if a node does not detect the event, the sensor reading is not above the threshold, and the output is '0'. A faulty node outputs the opposite of the truth, that is a false negative or a false positive respectively. In addition, only events of a reasonable size are considered, spanning at least a couple of nodes.

We also assume that each node knows its one-hop-neighbours. This knowledge is established in an initialization phase. WSNs are characterized by their severe restraints on energy, and unfortunately, radio communication is rather power-hungry. We therefore consider WSNs characterized by a high quantity of node-to-node communication. In such systems, the data required by our algorithm can be piggy-backed on data packets of the regular data flow. In such scenarios, we assume that reasonable energy supplies are provided in the form of large energy reservoirs and/or energy harvesting devices. Standard batteries can also be a sufficient energy supply for WSNs with a shorter mission time.

C. Fault Diagnosis - by Means of Playing Advanced Minesweeping

Each node has a total of 8 neighbours. The node and each of its neighbours either detect the event or not. Nodes detecting an event are displayed dark in figures. The digits at each node refer to the number of neighbouring nodes that detect an event. The consideration of these numbers was inspired by the Minesweeper game, where digits show the number of mines in the adjacent fields. A mine in Minesweeper corresponds to a neighbouring node that detects an event in our algorithm.

Table I summarizes the nomenclature used. The nomenclature consists of terms adopted from [7] complemented with terms newly introduced for our algorithm.

Definition A *positive* node or neighbour refers to a node that detects an event.

A node is faulty if the binary variable denoting the presence of an event at node a, T_A, is complementary to the binary variable S_a that indicates if an event has been detected by node a: $T_a \neq S_a$

Every node is assigned a value $v(a)$, characterizing the node's inclination to be faulty. If $v(a)$ is larger than a threshold Θ, the node is believed not to have detected the event, resulting in R_a = 1, otherwise R_a = 0. A fault was found by the diagnosis algorithm if $T_a \neq S_a$ and $R_a = T_a$

Table I
SUMMARY OF THE USED NOMENCLATURE

Symbol	Definition
a	The currently considered node
b_i	A node adjacent to node a; $\{B\}$ is the set of all b_i
c_i	A node adjacent to node a that is detecting an event; $\{C\}$ is the set of all c_i
T_a	Binary variable indicating if an event is present at node a
S_a	Binary variable indicating the detection result of node a
R_a	Binary variable indicating if the node considers to have detected an event after the diagnosis algorithm has been run
N	The number of neighbours of node a that detect an event
$P(all)$	The number of all neighbours' adjacent nodes that detect an event
$P(N)$	Same as P(all), but considering only positive neighbours of node a
Dom(a)	The maximum number of positive neighbours any positive node adjacent to node a has
$\omega(a)$	The weighting factor $\omega(a) = \frac{8-Dom(a)}{N}$
Θ	Threshold
$v(a)$	The decision value compared to Θ

and a newly introduced fault corresponds to $T_a = S_a$ and $R_a \neq T_a$.

1) Computation of the Decision Value $v(a)$: The set $\{B\}$ contains all nodes b_i that are adjacent to the node a. The set $\{C\}$ contains all nodes c_i, that are adjacent to the node a, with S_{c_i} = 1 $\rightarrow$ $\{C\} \subset \{B\}$. The number of neighbours of a node x that detect an event is generally computed by

$$positives(a) = |C| = N$$

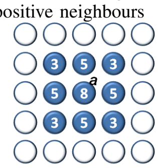
Computing

$$P(all) = \sum_{i=1}^{|B|} positives(b_i)$$
$$P(N) = \sum_{i=1}^{|C|} positives(c_i)$$
$$Dom(a) = max_{c_i \in C}\big(positives(c_i)\big)$$
$$w = \frac{8 - Dom(a)}{N}$$

leads to the decision value $v(a)$

$$v(a) = \begin{cases} \omega(a) \cdot \frac{P(N)}{P(all)} & P(N) \neq 0 \\ \infty & P(N) = 0 \end{cases}$$

Figure 1. $Dom(a)$ = 8 for all neighbours of the central node a, because a has 8 positive neighbours



finally resulting in the decision R_a

$$R_a = \begin{cases} 1 & v(a) < \Theta \\ 0 & otherwise \end{cases}$$

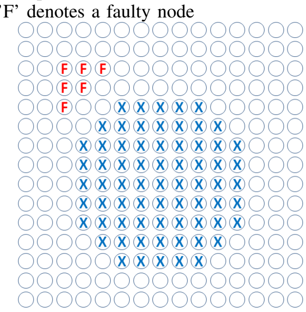
$P(N)$ yields zero if N = 0 that is, no neighbour detects an event. This means that the node is the only one detecting the event within a one-hop-radius. Assigning an infinitely large $v(a)$ value to a node entails a guaranteed exceeding of the threshold. Therefore, the node is believed not to have detected an event, agreeing with our precondition about minimal event size.

The other extreme case is $Dom(a)$ = 8, as this implies $\omega(a)$ = 0, and consequently $v(a)$ = 0. $Dom(a)$ being maximal means that there is at least one node among the elements of $\{C\}$ that features 8 positive neighbours. Note that this can also refer to the node a itself. Assuming an error free scenario, a node having 8 positive neighbours is situated within the event region, at least one hop away from the boundary. The weighting scheme considers nodes in such a position as part of the event region. In the scenario depicted in Figure 1, dark nodes sense the event, while white nodes do not. All neighbours of node a are assigned $\omega(a)$ = 0, because for all neighbours x_i applies $Dom(x_i)$ = 8. Note that a node can only be assigned $Dom(a)$ = 8 if the node itself detects an event. The depicted scenario has the minimum number of nodes, where assigning $\omega(a)$ = 0 is possible.

2) Threshold Selection: Our diagnosis algorithm is optimised for the detection of convex event regions. Stating a minimum number of positive nodes necessary in the vicinity is not meaningful. This is because $v(a)$ generally is influenced by the number of detections in a two-hop-radius. In addition, we do not target small events that are detected by less than seven nodes. We found that a threshold Θ = 0.1775 performs well in experiments with convex event regions.

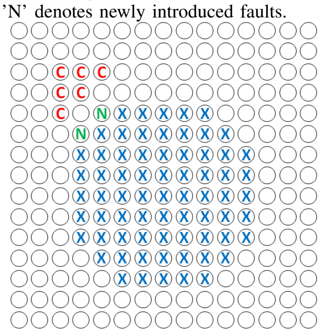
3) Sample Scenario: Figure 2 shows a scenario with a circular event region at the centre and six faulty nodes in the top left. An event is present at nodes represented by an 'X' and nodes outside the event region are denoted by an 'O'. Faulty nodes are depicted by an 'F'.

Figure 2. Sample scenario: 'X' denotes a node that detects an event and 'F' denotes a faulty node



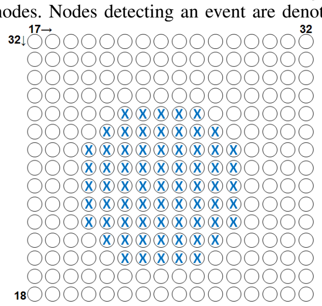
The corresponding output of our diagnosis algorithm is depicted in Figure 3.

Figure 3. Diagnosis for the sample scenario in Figure 2 obtained from our algorithm. 'C' denotes correctly identified faults and 'N' denotes newly introduced faults.



The algorithm correctly identifies all six faulty nodes (indicated as 'C'), but introduces two new faults (indicated as 'N'). This happens because the two newly introduced faults n_1 and n_2 are situated between two clusters of positive nodes. The larger cluster consists of nodes that really detect an event

Figure 4. The evaluation scenario similar to the one in [7] with only the event region in the upper right quadrant shown. The numbers indicate the column and line with the origin located in the lower left corner, the entire evaluated WSN grid consists of 32x32 nodes. Nodes detecting an event are denoted by 'X'



and the smaller cluster consists of nodes that incorrectly report to have detected an event. The diagnosis algorithm does not distinguish between 'detecting' and 'believing to detect'. Nodes n_1 and n_2 are considered to be faulty because $v(n_1)$ = 0.146 and $v(n_1)$ = 0.150 which is relatively close to the threshold Θ = 0.1775

IV. Experimental Results

To evaluate the performance of our algorithm, we created a scenario equivalent to the one used in [7]. For better comparability, we adopted the metrics of evaluation from [7] and [3] respectively.

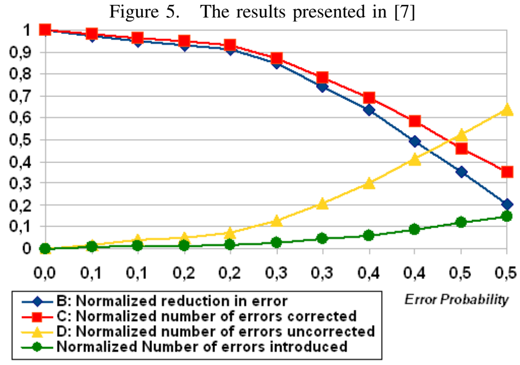
A. Setup and Experiment

We have developed a Java application for the simulation and evaluation of our algorithm. We simulate a WSN of dimension 32x32 nodes with the nodes arranged in a grid topology, as depicted in Figure 4. As before, nodes, where the event is present are represented by an 'X', nodes outside the event region are denoted by an 'O'. The circular event region with a diameter of 9 nodes is situated in the upper right quadrant. Starting with an initial error rate of p_{err} = 0.0 we increase p_{err} in steps of 0.05 up to a maximum of p_{err} = 0.5. which corresponds to every second node being faulty. We performed 100 simulation runs for every error rate in the interval [0,0.5] and averaged the individual results. The threshold was set to Θ = 0.1775.

Table II
Employed metrics, adapted from [3] and [7]

Symbol	Definition	Evaluation Function
α	Average number of errors after decoding	B = $\frac{1-\alpha}{np}$
β	Average number of errors corrected	C = $\frac{\beta}{np}$
γ	Average number of errors uncorrected	D= $\frac{\gamma}{np}$
δ	Average number of new errors introduced	E = $\frac{\delta}{np}$

Figure 5. The results presented in [7]



B. Evaluation

Table II lists the employed metrics. We compare our results only to the findings of [7], displayed in Figure 5 as the results for all evaluation functions in [7] are better than those presented in [3]. The graphs plotted in Figure 6 illustrate the performance of our approach. Figure 7 compares the average number of errors after decoding (Evaluation Function 'B') of our algorithm to the results presented in [7]. This metric considers undetected errors as well as newly introduced errors. For lower error probabilities, our algorithm performs slightly worse, but for $p_{err} > 0.25$ our algorithm performs considerably better. The strength of our algorithm is that the number of corrected errors is nearly constant and the number of newly introduced errors and uncorrected errors is low. In order to mistake a fault-free node for a faulty one, there has to be a larger accumulation of nodes that detect an event,

Figure 6. The results achieved by our algorithm.

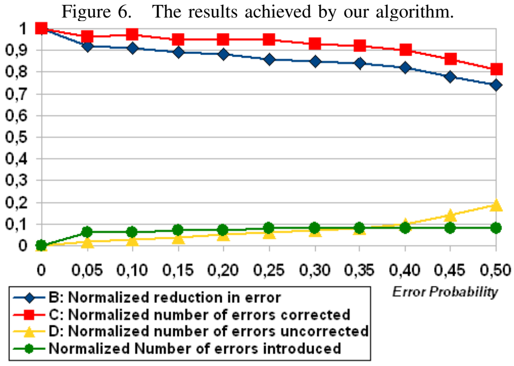
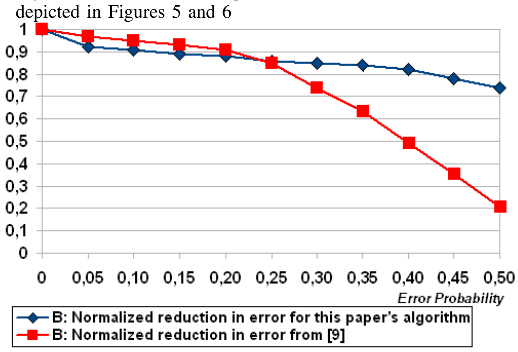


Figure 7. Exclusive comparison of Evaluation Function B depicted in Figures 5 and 6



no matter if they are right or not. This effect is illustrated by Figure 2.

We have learned from previous experiments that algorithms that perform very well at detecting errors tend to introduce a lot of new errors, yielding a large total of errors after decoding. We believe that our approach shows a good balance between the amount of detected errors and newly introduced errors.

For an even improved performance we plan to explore a hybrid scheme that combines these two approaches into one algorithm. As the nodes are not aware of the actual error rate or the number of detected errors, this scheme would require a supervising instance, e.g. a clusterhead. This clusterhead would be aware of the number of detected errors within its cluster and would select an algorithm depending on the number of detected errors.

V. Conclusion and Future Work

We presented a dependability concept targeting WSNs that perform event detection. Our considerations concentrated on furnishing nodes with means to perform fault diagnosis autonomously, this being one step towards attaining dependability. The key feature was the comparison of the findings of the considered node, the node's neighbours and also the findings of the nodes adjacent to the neighbours. We showed that our algorithm performs nearly as good as the one presented in [7] for an error probability of $p_{err} < 0.25$, but performs considerably better for higher error probabilities of up to p=0.5.

Further work will include exploring ways to describe event characteristics such as non-convexity by processable statements from which optimal thresholds can be deducted.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38(4), 2002.

[2] D. M. Doolina and N. Sitara. Wireless sensors for wildire monitoring. In *SPIE Symposium on Smart Structures & Materials NDE 2005*, March 2005.

[3] B. Krishnamachari and S. Iyengar. Distributed bayesian algorithms for fault-tolerant event region detection in wireless sensor networks. In *IEEE Transactions on Computers*, volume 53, 2004.

[4] J. Kumagi. Life of birds. *IEEE Spectrum*, 41, 2004.

[5] E. Ould-Ahmed-Vall, G. F. Riley, and B. S. Heck. A distributed fault-tolerant algorithm for event detection using heterogeneous wireless sensor networks. In *45th IEEE Conference on Decision and Control*, 2006.

[6] G. Werner-Allen, K. Lorincz, M. Welsh, M. Ruiz, O. Marcillo, J. Johnson, and J. Lees. Deploying a wireless sensor network on an active volcano. *IEEE Internet Computing*, 10, March-April 2006.

[7] Y. Xiang, H. Li, Z. Xie, and P. Li. Distributed weighting fault-tolerant algorithm for even region detection in wireless sensor networks. In *International Conference on Communications, Circuits and Systems*, 2008.

Bibliography

[ACGV11] Giuseppe Amato, Stefano Chessa, Claudio Gennaro, and Claudio Vairo. Efficient Detection of Composite Events in Wireless Sensor Networks: Design and Evaluation. In *Computers and Communications (ISCC), 2011 IEEE Symposium on*, pages 821–823, 2011.

[AGPB+12] Alvaro Araujo, Jaime García-Palacios, Javier Blesa, Francisco Tirado, Elena Romero, Avelino Samartín, and Octavio Nieto-Taladriz. Wireless Measurement System for Structural Health Monitoring With High Time-Synchronization Accuracy. *IEEE Transactions on Instrumentation and Measurement*, 61(3):801–810, March 2012.

[ALRL04] Algirdas Avižienis, Jean-Claude Laprie, Brian Randell, and Carl Landwehr. Basic Concepts and Taxonomy of Dependable and Secure Computing. *IEEE Transactions on Dependable and Secure Computing*, 01(1):11–33, 2004.

[Ant13] Gary Anthes. Inexact Design: Beyond Fault-Tolerance. *Communications of the ACM*, 56(4):18–20, 2013.

[ASSC02] Ian F. Akyildiz, Wei Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. Wireless Sensor Networks: A Survey. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 38(4):393–422, March 2002.

[Avi13] Algirdas Avižienis. The Architecture of a Resilience Infrastructure for Computing and Communication Systems. In *Dependable Systems and Networks (DSN), 2013 43rd Annual IEEE/IFIP International Conference on*, pages 1–2, 2013.

[BBH+10] Carlo A. Boano, James Brown, Zhitao He, Utz Roedig, and Thiemo Voigt. Low-Power Radio Communication in Industrial Outdoor Deployments: The Impact of Weather Conditions and ATEX-Compliance. In *Sensor Applications, Experimentation, and Logistics*, volume 29 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 159–176. Springer Berlin Heidelberg, 2010.

[BGG08] Kenneth Bannister, Gianni Giorgetti, and Sandeep K. S. Gupta. Wireless Sensor Networking for "Hot" Applications: Effects of Temperature on Signal Strength, Data Collection and Localization. In *Proceedings of the fifth Workshop on Embedded Networked Sensors*, June 2008.

[BISV08] Guillermo Barrenetxea, François Ingelrest, Gunnar Schaefer, and Martin Vetterli. The Hitchhiker's Guide to Successful Wireless Sensor Network Deployments. In *Proceedings of the 6th ACM Conference on Embedded Networked Sensor Systems (SenSys08)*, pages 43–56, 2008.

[BKM+12] Nouha Baccour, Anis Koubâa, Luca Mottola, Marco Antonio Zú niga, Habib Youssef, Carlo Alberto Boano, and Mário Alves. Radio Link Quality Estimation in Wireless Sensor Networks: A Survey. *ACM Transactions on Sensor Networks*, 8(4), September 2012.

[BMH09] Majid Bahrepour, Nirvana Meratnia, and Paul J. M. Havinga. Sensor Fusion-based Event Detection in Wireless Sensor Networks. In *Mobile and Ubiquitous Systems: Networking Services, MobiQuitous, 2009. MobiQuitous '09. 6th Annual International*, pages 1–8, 2009.

[BMH11] Majid Bahrepour, Nirvana Meratnia, and Paul J. M. Havinga. Online Unsupervised Event Detection in Wireless Sensor Networks. In *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2011 Seventh International Conference on*, pages 306–311, 2011.

[BMP+10] Majid Bahrepour, Nirvana Meratnia, Mannes Poel, Zahra Taghikhaki, and Paul J. M. Havinga. Distributed Event Detection in Wireless Sensor Networks for Disaster Management. In *Intelligent Networking and Collaborative Systems (INCOS), 2010 2nd International Conference on*, pages 507–512, 2010.

[BTV+10] Carlo A. Boano, Nicolas Tsiftes, Thiemo Voigt, James Brown, and Utz Roedig. The Impact of Temperature on Outdoor Industrial Sensornet Applications. *Industrial Informatics, IEEE Transactions on*, 6(3):451–459, 2010.

[BVN+11] Carlo A. Boano, Thiemo Voigt, Claro Noda, Kay Römer, and Marco Zuniga. JamLab: Augmenting Sensornet Testbeds with Realistic and Controlled Interference Generation. In *Information Processing in Sensor Networks (IPSN), 2011 10th International Conference on*, pages 175–186, 2011.

[CB10] Marcus Chang and Philippe Bonnet. Monitoring in a High-Arctic Environment: Some Lessons from MANA. *IEEE Pervasive Computing*, 9(4):16–23, October 2010.

[CCM+09] Marcello Cinque, Domenico Cotroneo, Catello Di Martino, Stefano Russo, and Alessandro Testa. AVR–INJECT: A Tool for Injecting

Faults in Wireless Sensor Nodes. In *23rd IEEE International Symposium on Parallel and Distributed Processing*, pages 1–6, 2009.

[CLF05] Qingchun Chen, Kam-Yiu Lam, and Pingzhi Fan. Comments on "Distributed Bayesian Algorithms for Fault-Tolerant Event Region Detection in Wireless Sensor Networks". *IEEE Transactions on Computers*, 54(9):1182–1183, 2005.

[CLL+08] Joyce Coleman, Tony Lau, Bhushan Lokhande, Peter Shum, Robert Wisniewski, and Mary Peterson Yost. The Autonomic Computing Benchmark. In *Dependability Benchmarking for Computer Systems*, pages 1–21. Wiley & Sons, Inc, 2008.

[CMT12] Marcello Cinque, Catello Di Martino, and Alessandro Testa. Analyzing and Modeling the Failure Behavior of Wireless Sensor Networks Software under Errors. In *8th IEEE International Wireless Communications and Mobile Computing Conference*, pages 567–569. IEEE, 2012.

[Con05] Cristian Constantinescu. Dependability Benchmarking Using Environmental Test Tools. In *Annual Reliability and Maintainability Symposium*, pages 567–571, 2005.

[Con13] RIPLECS Project Consortium. RIPLECS Project. `http://riplecs.dipseil.net/`, 2011-2013. Last accessed: 2014-01-17.

[CRS99] Joao Carlos Cunha, Mário Zenha Rela, and Joao Gabriel Silva. Can Software Implemented Fault-Injection be Used on Real-Time Systems? In *Dependable Computing*, volume 1667 of *Lecture Notes in Computer Science*, pages 209–226. Springer Berlin Heidelberg, 1999.

[CWJ+10] Peter Corke, Tim Wark, Raja Jurdak, Wen Hu, Philip Valencia, and Darren Moore. Environmental Wireless Sensor Networks. *Proceedings of the IEEE*, 98(11):1903–1917, 2010.

[dAFRG09] David de Andrés, Jesus Friginal, Juan Carlos Ruiz, and Pedro J. Gil. An Attack Injection Approach to Evaluate the Robustness of Ad Hoc Networks. In *15th IEEE Pacific Rim International Symposium on Dependable Computing*, pages 228–233. IEEE Computer Society, 2009.

[DEM+12] Vladimir Dyo, Stephen A. Ellwood, David W. Macdonald, Andrew Markham, Niki Trigoni, Ricklef Wohlers, Cecilia Mascolo, Bence Pásztor, Salvatore Scellato, and Kharsim Yousef. WILDSENSING: Design and Deployment of a Sustainable Sensor Network for Wildlife Monitoring. *ACM Transactions on Sensor Networks*, 8(4):29:1–29:33, September 2012.

[dQMS10] Diego de Queiroz Macedo and Jaime S. Sichman. Analysis of Von Neumann Neighborhoods in Parallel Multi-agent Simulations. In *Second Brazilian Workshop on Social Simulation*, pages 27–32, Oct 2010.

[EPF13] EPFL. Sensorscope: Sensor Networks for Environmental Monitoring. `http://infoscience.epfl.ch/record/180186?ln=en`, 2013. Last accessed: 2013-12-12.

[FdARG11] Jesus Friginal, David de Andrés, Juan-Carlos Ruiz, and Pedro Gil. Using Performance, Energy Consumption, and Resilience Experimental Measures to Evaluate Routing Protocols for Ad Hoc Networks. In *Network Computing and Applications (NCA), 2011 10th IEEE International Symposium on*, pages 139–146, 2011.

[FdARM11] Jesus Friginal, David de Andrés, Juan-Carlos Ruiz, and Regina Moraes. Using Dependability Benchmarks to Support ISO/IEC SQuaRE. In *Proceedings of the 2011 IEEE 17th Pacific Rim International Symposium on Dependable Computing*, pages 28–37. IEEE Computer Society, 2011.

[FEDV08] Niclas Finne, Joakim Eriksson, Adam Dunkels, and Thiemo Voigt. Experiences from Two Sensor Network Deployments Self-Monitoring and Self-Configuration Keys to Success. In *Wired/Wireless Internet Communications*, volume 5031 of *Lecture Notes in Computer Science*, pages 189–200. Springer Berlin Heidelberg, 2008.

[GST$^+$08] Jayant Gupchup, Abhishek Sharma, Andreas Terzis, Al Burns, and Alex Szalay. The perils of detecting measurement faults in environmental monitoring networks. In *IEEE International Conference on Distributed Computing in Sensor Systems*, 2008.

[HC02] Jason L. Hill and David E. Culler. Mica: A Wireless Platform for Deeply Embedded Networks. *Micro, IEEE*, 22(6):12–24, 2002.

[HDCJ12] Wen Hu, Tuan Le Dinh, Peter Corke, and Sanjay Jha. Outdoor Sensornet Design and Deployment: Experiences from a Sugar Farm. *Pervasive Computing, IEEE*, 11(2):82–91, 2012.

[HHW12] Karima B. Hein, Leander B. Hörmann, and Reinhold Weiß. Using a Leaky Bucket Counter as an Advanced Threshold Mechanism for Event Detection in Wireless Sensor Networks. In *Proceedings of the Tenth Workshop on Intelligent Solutions in Embedded Systems*, pages 51–56, 2012.

[HHW13] Karima B. Hein, Leander B. Hörmann, and Reinhold Weiß. Analysis of Threshold-Based Event Detection Algorithms for Wireless Sensor Networks by Fault Injection. In *Ubiquitous Intelligence and Computing, 2013 IEEE 10th International Conference on and 10th International*

Conference on Autonomic and Trusted Computing (UIC/ATC), pages 662–667. IEEE Computer Society, December 2013.

[HSKK12] Leander B. Hörmann, Michael Steinberger, Michael Kalcher, and Christian Kreiner. Educational Remote Lab Concept for Energy Harvesting Enhanced Wireless Sensor Networks. In *5th European DSP Education and Research Conference (EDERC)*, pages 95–99, 2012.

[HSKK13] Leander B. Hörmann, Michael Steinberger, Michael Kalcher, and Christian Kreiner. Using a Remote Lab for Teaching Energy Harvesting Enhanced Wireless Sensor Networks. In *Global Engineering Education Conference (EDUCON), 2013 IEEE*, pages 1109–1117, 2013.

[HSX+12] Renjie Huang, Wen-Zhan Song, Mingsen Xu, Nina Peterson, Behrooz Shirazi, and Richard LaHusen. Real-World Sensor Network for Long-Term Volcano Monitoring: Design and Findings. *IEEE Transactions on Parallel and Distributed Systems*, 23(2):321–329, Feb 2012.

[HTI97] Mei-Chen Hsueh, Timothy K. Tsai, and Ravishanar K. Iyer. Fault injection techniques and tools. *Computer*, 30(4):75–82, 1997.

[HV11] Zhitao He and Thiemo Voigt. Precise Packet Loss Pattern Generation by Intentional Interference. In *Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011 International Conference on*, pages 1–6, 2011.

[HW09] Karima B. Hein and Reinhold Weiß. Minesweeper for sensor networks – making event detection in sensor networks dependable. In *International Conference on Computational Science and Engineering 2009*, volume 1, pages 388–339. IEEE Computer Society, August 2009.

[IBS+10] François Ingelrest, Guillermo Barrenetxea, Gunnar Schaefer, Martin Vetterli, Olivier Couach, and Marc Parlange. SensorScope: Application-specific Sensor Network for Environmental Monitoring. *ACM Transactions on Sensor Networks*, 6(2):17:1–17:32, March 2010.

[IEC13] IEC. *International Electrotechnical Vocabulary. Chapter 191: Dependability and quality of service*, 1990–2013.

[Inc07] Crossbow Technology Inc. MTS/MDA Sensor Board Users Manual, Revision A. `http://www.memsic.com/userfiles/files/Datasheets/WSN/mts_mda_datasheet.pdf`, 2007. Last accessed: 2014-05-07.

[Inc14a] MEMSIC Inc. MICAz Wireless Measurement System. `www.memsic.com/userfiles/files/Datasheets/WSN/micaz_datasheet-t.pdf`, 2014. Last accessed: 2014-05-07.

[Inc14b] MEMSIC Inc. MTS/MDA Sensor, Data Acquisition Boards. `www.memsic.com/userfiles/files/Datasheets/WSN/micaz_datasheet-t.pdf`, 2014. Last accessed: 2014-01-22.

[ISO10a] ISO/IEC. *Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – Evaluation module for recoverability*, 2010.

[ISO10b] ISO/IEC. *Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models*, 2010.

[KI04] Bhaskar Krishnamachari and Sitharama Iyengar. Distributed Bayesian Algorithms for Fault-Tolerant Event Region Detection in Wireless Sensor Networks. *IEEE Transactions on Computers*, 53(3):241–250, 2004.

[KPSV02] Farinaz Koushanfar, Miodrag Potkonjak, and Alberto Sangiovanni-Vincentell. Fault Tolerance Techniques for Wireless Ad Hoc Sensor Networks. In *Sensors 2002, Proceedings of IEEE*, volume 2, pages 1491–1496, June 2002.

[KPSV03] Farinaz Koushanfar, Miodrag Potkonjak, and Alberto Sangiovanni-Vincentelli. On-line Fault Detection of Sensor Measurements. In *IEEE Sensors*, volume 2, pages 974 – 979, 2003.

[KS08] Karama Kanoun and Lisa Spainhower. *Dependability Benchmarking for Computer Systems*. Wiley & Sons, Inc, 2008.

[Lap08] Jean-Claude Laprie. From Dependability to Resilience. In *38th IEEE/IFIP Int. Conf. On Dependable Systems and Networks*, 2008.

[LAV+10] Yingshu Li, Chunyu Ai, Chinh T. Vu, Yi Pan, and Raheem Beyah. Delay-Bounded and Energy-Efficient Composite Event Monitoring in Heterogeneous Wireless Sensor Networks. *Parallel and Distributed Systems, IEEE Transactions on*, 21(9):1373–1385, 2010.

[LBV06] Koen Langendoen, Aline Baggio, and Otto Visser. Murphy Loves Potatoes: Experiences from a Pilot Sensor Network Deployment in Precision Agriculture. In *International Workshop on Parallel and Distributed Real-Time Systems (WPDRTS)*, 2006.

[LCZ09] Steven Lai, Jiannong Cao, and Yuan Zheng. PSWare: A Publish / Subscribe Middleware Supporting Composite Event in Wireless Sensor Network. In *IEEE International Conference on Pervasive Computing and Communications*, pages 1–6, 2009.

[LYC09] Myeong-Hyeon Lee, Sung-Jib Yim, and Yoon-Hwa Choi. Grid-based Fault-Tolerant Event Detection in Wireless Wensor Networks. In *TENCON 2009 - 2009 IEEE Region 10 Conference*, pages 1–5, 2009.

[Mai10] Philipp Maierl. SeNetSim - Simulator für Dependability-Betrachtungen für Event-Detection in Sensornetzwerken. Bachelor's Thesis, Institute for Technical Informatics, Graz University of Technology, 2010.

[MCC12] Catello Di Martino, Marcello Cinque, and Domenico Cotroneo. Automated Generation of Performance and Dependability Models for the Assessment of Wireless Sensor Networks. *Computers, IEEE Transactions on*, 61(6):870–884, 2012.

[MCP+02] Alan Mainwaring, David Culler, Joseph Polastre, Robert Szewczyk, and John Anderson. Wireless sensor networks for habitat monitoring. In *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 88–97. ACM Press, 2002.

[MMA05] H. Morikawa M. Minami, T. Morito and T. Aoyama. Solar Biscuit: A Battery-less Wireless Sensor Network System for Environmental Monitoring Applications. In *Proceedings of the 2nd International Workshop on Networked Sensing Systems*, pages 51–56, 2005.

[MR13] Dorothy N. Monekosso and Paolo Remagnino. Data reconciliation in a smart home sensor network. *Expert Systems with Applications*, 40(8):3248 – 3255, 2013.

[MS06] Fernando Martincic and Loren Schwiebert. Distributed Event Detection in Sensor Networks. In *International Conference on Systems and Networks Communications*, 2006.

[PWC+11] W.-B. Pottner, L. Wolf, J. Cecilio, P. Furtado, R. Silva, J.S. Silva, A. Santos, P. Gil, A. Cardoso, Z. Zinonos, J. do O, B. McCarthy, J. Brown, U. Roedig, T. O'Donovan, C. J. Sreenan, Z. He, T. Voigt, and A. Jugel. WSN Evaluation in Industrial Environments First Results and lessons learned. In *Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011 International Conference on*, pages 1–8, 2011.

[RCK+05] Nithya Ramanathan, Kevin Chang, Rahul Kapur, Lewis Girod, Eddie Kohler, and Deborah Estrin. Sympathy for the Sensor Network Debugger. In *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems*, SenSys '05, pages 255–267. ACM, 2005.

[SCL08] Byunghun Song, Haksoo Choi, and Hyung Su Lee. Surveillance Tracking System Using Passive Infrared Motion Sensors in Wireless Sensor Network. In *International Conference on Information Networking*, pages 1–5, January 2008.

[SM14] Samrat Sarkar and Koushik Majumder. A Survey on Power Aware Routing Protocols for Mobile Ad-Hoc Network. In *Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2013*, volume 247 of *Advances in Intelligent Systems and Computing*, pages 313–320. Springer International Publishing, 2014.

[Som12] Ian Sommerville. *Software Engineering*. Pearson Studium, Ninth edition, 2012.

[SPMC04] Robert Szewczyk, Joseph Polastre, Alan Mainwaring, and David Culler. Lessons From A Sensor Network Expedition. In *European Conference on Wireless Sensor Networks (EWSN)*, pages 307–322, 2004.

[SS92] Daniel P. Siewiorek and Robert S. Swarz. *Reliable computer systems - design and evaluation (2. ed.)*. Digital Press, 1992.

[Tan96] Andrew S. Tanenbaum. *Computer Networks*. Prentice Hall Professional Technical Reference, 3rd edition, 1996.

[Top14] Top500.org. About The Linpack Benchmark. `http://www.top500.org/project/linpack/`, 2014. Last accessed: 2014-01-15.

[Tur86] Jonathan S. Turner. New directions in communications (or which way to the information age?). *IEEE Communications Magazine*, 24(10):8–15, October 1986.

[WBL12] Matthias Woehrle, Martin Bor, and Koen Langendoen. 868 MHz: A noiseless environment, but no free lunch for protocol design. In *2012 Ninth International Conference on Networked Sensing Systems (INSS)*, pages 1–8, 2012.

[YC09] Sung-Jib Yim and Yoon-Hwa Choi. Fault-Tolerant Event Detection Using Two Thresholds in Wireless Sensor Networks. In *15th IEEE Pacific Rim International Symposium on Dependable Computing*, pages 331 – 335, November 2009.

[ZMH10] Yang Zhang, Nirvana Meratnia, and Paul Havinga. Outlier Detection Techniques for Wireless Sensor Networks: A Survey. *Communications Surveys Tutorials, IEEE*, 12(2):159–170, April 2010.

[ZR07] Michael Zoumboulakis and George Roussos. Escalation: Complex Event Detection in Wireless Sensor Networks. In *Smart Sensing and Context*, volume 4793 of *Lecture Notes in Computer Science*, pages 270–285. Springer Berlin Heidelberg, 2007.

[ZR09] Michael Zoumboulakis and George Roussos. Efficient Pattern Detection in Extremely Resource-Constrained Devices. In *Sensor, Mesh and*

Ad Hoc Communications and Networks, SECON '09. 6th Annual IEEE Communications Society Conference on, pages 1–9, 2009.