Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

9. Fachgespräch Sensornetze

der GI/ITG Fachgruppe Kommunikation und Verteilte Systeme

16.-17. September 2010

Institut für Informatik Technischer Bericht

Prof. Dr. Reiner Kolla (Hrsg.)

urn:nbn:de:bvb:20-opus-51106
http://www.opus-bayern.de/uni-wuerzburg/volltexte/2010/5110/

KuVS
GI ITG
innoventis

Lehrstuhl für Informatik V
Am Hubland
97074 Würzburg

UNIVERSITÄT WÜRZBURG
INSTITUT FÜR INFORMATIK

9. Fachgespräch "Sensornetze"

der GI/ITG Fachgruppe Kommunikation und Verteilte Systeme

Technischer Bericht
Reiner Kolla (Hrsg.)

16. - 17. September 2010

Inhaltsverzeichnis

Session 1

MAUS: A Multi-hop Autonomous Sensor Network for Monitoring Applications with Full IP-support **1**
Alexander Klein, Lothar Braun, Corinna Schmitt, Georg Carle (Technische Universität München)

Dynamic Memory Management for Resource Constrained Sensor/Actor Systems ... **5**
Marcel Baunach (University of Würzburg)

Smart Energy Module for Wireless Sensor Nodes **9**
Juergen Jessen, Marcus Venzke, Volker Turau (Hamburg University of Technology)

Session 2

Benchmarking of WSN Solutions and IEEE 802.15.4-2006 PSSS based Solutions ... **13**
Andreas C. Wolf (Dr. Wolf Wireless GmbH, Teltow), Matthias Mahlig (IHP GmbH Frankfurt/Oder)

Holistic Packet Statistics for Neighborhood Management in Sensor Networks **17**
Sebastian Ernst, Christian Renner, Christoph Weyer, Volker Turau (Hamburg University of Technology)

Desynchronization in Multi-Hop Topologies: A Challenge **21**
Clemens Mühlberger (University of Würzburg)

Poster Session

Describing Packet Payload Structures using Lightweight Semantic Data Type Annotations **25**
Andreas Reinhardt, Diego Costantini, Ralf Steinmetz (Technische Universität Darmstadt)

Fine-grained Access Control Enabling Privacy Support in Wireless Sensor Networks **29**
Delphine Christin, Andreas Reinhardt (Technische Universität Darmstadt), Salil S. Kanhere (University of New South Wales), Matthias Hollick (Technische Universität Darmstadt)

Senkung der Versicherungsprämien bei Überlandtransporten mittels geeigneter Smart Object Technologien **33**
Hauke Traulsen, Christopher Kaffenberger, Alexander Pflaum (Fraunhofer-Institut IIS, Fürth)

Wireless Energy Transmission for Implantable Wireless Sensor Nodes **37**
Tristan Bremer, Maike Vollmer (University of Würzburg)

Session 3

Tuontu: A Tool for Evaluating the Impact of Wireless Sensor Network Design Alternatives **39**
Barbara Staehle, Markus Leimbach, Dirk Staehle (University of Würzburg)

T-SIM: A Simulation Enviroment for Dynamic Wireless Sensor Networks **43**
Christian Huisinga, Jens Kamenik (OFFIS Oldenburg),
Axel Hahn (Carl von Ossietzky University of Oldenburg)

An Algorithm for Fast Symmetry Reduction in Symbolic Model Checking **47**
Christian Appold (University of Würzburg)

Session 4

Mobility Assisted Positioning in Wireless Sensor Networks **51**
Hannes Frey, Martin Schwier (University of Paderborn)

Exploiting Semantic Quorum-Based Data Replication in Wireless Sensor Networks **55**
Kinga Kiss Iakab, Felix Jonathan Oppermann, Oliver Theel
(Carl von Ossietzky University of Oldenburg), Jens Kamenik (OFFIS Oldenburg)

Exploring the Applicability of Participatory Sensing in Emergency Scenarios **59**
Diego Costantini, Andreas Reinhardt, Parag S. Mogre, Ralf Steinmetz
(Technische Universität Darmstadt)

Session 5

A Report on System and Radio Transmission Issues in a Star-Topology WSN **63**
Sebastian A. Bachmaier (Universität Stuttgart)

Deployment of Wireless Sensor Networks in Logistics – Potential, Requirements, and a Testbed **67**
Sebastian Zöller, Andreas Reinhardt, Marek Meyer, Ralf Steinmetz
(Technische Universität Darmstadt)

Über die Notwendigkeit einer Integrationsplattform für unterschiedliche Smart Object Technologien **71**
Sebastian Lempert, Alexander Pflaum (Fraunhofer-Institut IIS, Fürth)

Deterministic technique for energy-efficient centralized clustering of wireless sensor networks **75**
V. Delport, M. Gessner, T.D. Grossman, A. Singer
(University of Applied Sciences Mittweida)

Teilnehmerliste

Appold	Christian	Universität Würzburg
Arratia-Scheit	Elsy	Fraunhofer IIS, Fürth
Bachmaier	Sebastian	Universität Stuttgart
Bauer	Norbert	innoventis GmbH, Würzburg
Baunach	Marcel	Universität Würzburg
Braun	Lothar	TU München
Bregenzer	Jürgen	Universität Würzburg
Bremer	Tristan	Universität Würzburg
Costantini	Diego	TU Darmstadt
Delphine	Christin	TU Darmstadt
Delport	Volker	Fachhochschule Mittweida
Engel	Pascal	Fraunhofer ISC, Würzburg
Englert	Holger	innoventis GmbH, Würzburg
Frey	Hannes	Universität Paderborn
Geßner	Mario	Fachhochschule Mittweida
Huisinga	Christian	OFFIS Oldenburg
Jessen	Jürgen	TU Hamburg
Kaffenberger	Christopher	Fraunhofer IIS, Fürth
Kamenik	Jens	OFFIS Oldenburg
Kiss Iakab	Kinga	Universität Oldenburg
Klein	Alexander	TU München
Kolla	Reiner	Universität Würzburg
Lempert	Sebastian	Fraunhofer IIS, Fürth
Mühlberger	Clemens	Universität Würzburg
Oppermann	Felix Jonathan	Universität Lübeck
Peter	Steffen	IHP GmbH, Frankfurt/Oder
Reinhardt	Andreas	TU Darmstadt
Renner	Christian	TU Hamburg
Schmitt	Corinna	TU München
Staehle	Barbara	Universität Würzburg
Wolf	Andreas	Dr. Wolf Wireless GmbH, Teltow
Zöller	Sebastian	TU Darmstadt

MAUS: A Multi-hop Autonomous Sensor Network for Monitoring Applications with Full IP-support

Alexander Klein, Lothar Braun, Corinna Schmitt and Georg Carle
Department of Computer Science
Network Architectures and Services
Technische Universität München, Germany
Email: {klein,braun,schmitt,carle}@net.in.tum.de

I. Introduction

The majority of Wireless Sensor Network (WSN) solutions apply a simple one hop star topology since multi-hop communication has high demands on the communication protocols in terms of medium access and bandwidth consumption. Moreover, many solutions neglect security issues which might be acceptable for environment monitoring applications but are not a considerable choice for home (monitoring) applications that often transport confidential information. In addition, WSNs should support flexible mechanisms in order to exchange information in an efficient way. Efficient communication requires bidirectional communication between the sensor nodes and more powerful gateway nodes which provide connectivity to the Internet. Seamless connectivity between sensor nodes and computers in the Internet will become more important in next generations WSNs as the number of intelligent home automation networks is steadily increasing.

In recent years, new home infrastructures were introduced that focus on intelligent home scenarios which make use of WSNs. In our department, an Autonomic Home Networking Infrastructure (AutHoNe) [1] was developed. AutHoNe itself works with predefined knowledge to coordinate and manage different functionalities in the home to raise the resident's living comfort. The home automation system provides different functions such as remote access, knowledge sharing and security, aas well as a mechanism for the integration of new devices. This mechanism is used to integrate sensor nodes into the home infrastructure. We applied a modified version of the IPFIX protocol in order to enable flexible standardized way to collect information from the sensor nodes. Furthermore, we optimized the IPFIX protocol in terms of overhead by reducing the IPFIX protocol header to the absolute necessary. The information which we collected by using our compressed IPFIX (cIPFIX) [2], [3] solution was then transformed such that they could be integrated in the AutHoNe infrastructure.

II. Multi-hop AUtonomous Sensor Solution

The first results from the AutHone sensor extension encouraged us to go one step further by enhancing the functionality of the sensor network. For this reason, we started the development of the Multi-Hop Autonomous Sensor Solution (MAUS) system which is currently in a work in progress state since parts of its communication stack are still subject to modification. The most obvious limitation of many systems is represented by the star topology which makes the numerous deployments of base stations and/or more powerful gateway nodes necessary. The MAUS system provides full multi-hop functionality by integrating a modified version of the Statistic-Based-Routing (SBR) protocol [4] in the Tiny OS 2.1.1 [5] communication stack. In addition, we decided to use the Berkeley IP (BLIP) [6] implementation for low-power networks to enable seamless IP connectivity. The integration of IP increases the flexibility and simplifies the addressing of sensor nodes by administration nodes which are not part of the sensor network.

Due to the bidirectional communication, intelligent data gathering strategies can be applied to meet the requirements of the target application. The first results from the AutHoNe sensor extension have shown that cIPFIX provides efficient mechanisms to exchange information. Nonetheless, the current protocol version is designed as push protocol which does not allow reconfiguration of established data flows. Therefore, we are currently working on an extension of the cIPFIX protocol which supports bidirectional communication in order to allow a reconfiguration of the active flows, e.g. changing the push rate or the message format. In Section III, we briefly outline the changes of the cIPFIX protocol that are necessary to support this functionality.

The routing protocol is able to quickly detect link breaks without generating a large amount of overhead. Furthermore, it supports features, like delay-based forwarding, which can be used to either modify the topology or to balance the traffic load and thus the energy-consumption in the network. At the moment we are working on an integration of different mechanisms, e.g. the one which is presented in the latest version of the B.A.T.M.A.N. protocol [7], to improve the performance in multi-hop scenarios where a high percentage of links have asymmetric characteristics. A brief description of the protocol and the applied mechanism is given in Section IV. However, the performance of a wireless network strongly depends on the reliability of the underlying Medium Access Control (MAC) protocol.

Therefore, we decided to integrate the Backoff Preamble-based MAC protocol with sequential contention resolution (BPS-MAC) [8]. The protocol applies a new preamble-based access mechanism which combines ideas from the well-known

X-MAC protocol [9] and the Sift protocol [10]. The latter was introduced to minimize the collision probability of Carrier Sense Multiple Access (CSMA) based protocols by using a non-uniform distributed backoff duration. Sift chooses the distribution for the contention resolution with respect to the number of competing nodes which requires detailed knowledge of the network. The BPS-MAC protocol uses a similar mechanism to select the duration of the transmitted preambles which cover the task of reservation signals. The advantage of the mechanism which is applied by the BPS-MAC protocol is that it is less affected by the number of competing nodes and also addresses typical low-power transceiver related communication issues, like Clear Channel Assessment (CCA) delay and the turnaround time. The CCA delay represents the period of time which is required by the transceiver to detect a busy radio channel while the turnaround time specifies the period of time which is required to switch between receive and transmit mode. A description of the medium access procedure of the protocol is given in Section V.

Besides these communication related tasks, a sensor network, which is designed for autonomous home applications, has to deal with security issues that arise from decentralized communication. It is clear that sensor nodes cannot apply complex security strategies or provide a large number of keys since the computational power and the memory resources are very limited. Especially, the latter one has to be taken into account due to the fact that the routing table, the operating system, and the support of IP result in high memory consumption which limits the number of security solutions for our system. At the moment, we favor an approach which is based on the SPIN protocol [11].

III. Bidirectional IPFIX

IPFIX and its adoption for WSNs cIPFIX share two important properties: A simple and easy to extend information model and the separation of monitoring data and type information.

Data types in cIPFIX are represented using a simple (Type, Length) tuple, which allows the encoding of arbitrary sensor monitoring data. Type information is encoded in so called *Template* messages that are sent by the nodes to announce the type of data that they are going to export. As of now, cIPFIX is a unidirectional push protocol where sensor devices transmit their data to one or more collecting devices. A Template is sent from a sensor device to a collecting device after the sensor booted and before it starts to export monitoring data.

The collecting device, e.g. a central base station, has to receive and store the Template as it contains type information which is necessary to decode the monitoring data. After the Template has been transmitted, *Data Packets* can be exported. A node exports its data in periodic time intervals. Query messages that request data from the sensor devices are therefore not necessary.

The Data Packets contain sensor monitoring data without any further type information. Instead of type information, they carry a reference to the previously sent Template. A receiving collector can then decode the Data Packet using the referenced Template. Thus, the transmission efficiency is increased since type information has not to be included into every packet.

However, the current protocol faces some limitations at the moment: As the cIPFIX protocol is strictly unidirectional, the configuration of the Templates and the monitoring itself has to be conducted in a static way. This implies that a sensor device will always export all data from all monitoring data from the configured sensors, regardless of whether data export is necessary at a particular point in time. A node which is equipped with several sensors will always export data from all sensors, since it is not possible to change the Templates to contain only a specific subset of all available sensors.

In order to allow more flexibility, we plan to extend the existing protocol to contain more functionality. It is planned to convert the purely unidirectional push protocol into a bidirectional protocol, which allows the configuration of the sensor nodes. This can be achieved by extending the current protocol by new messages types that are derived from Templates.

Templates are used to announce the type of data which is exported from the node to the collecting process. Similar messages can be sent from the collector to the node in order to configure which data the node should export. This allows for the configuration of Templates on the devices. Using such messages, it is also possible to completely stop the export of sensor monitoring data. This can be done by configuring an empty Template on the sensor nodes, which will stop the export of data.

The cIPFIX protocol builds on IP and UDP and therefore suffers from packet loss, because UDP does not offer any reliability. Reliability for Template or configuration messages is a desired feature, as these messages occur rarely but are very important. It might be necessary to have the reliability feature for Data messages in some scenarios, too.

This feature has to be build on top of cIPFIX and can be build by adding another message type that acts as an acknowledgment messages. Reliability should be a per-message feature that can be set by the sending device if necessary. If a device considers a message to be important, it should set an acknowledgment-necessary flag which needs to be integrated into the cIPFIX message header. This flag triggers the transmission of an acknowledgment at the receiving device.

IV. Routing

The key characteristics of the SBR protocol are high end-to-end reliability (in fixed and mobile networks), load balancing capabilities, a smooth continuous routing metric, quick adaptation to changing network conditions, low processing and memory requirements, low overhead, support of unidirectional links and simplicity. The protocol can establish routes in a hybrid or a proactive mode and uses an adaptive continuous routing metric which makes it very flexible in terms of scalability while maintaining stable routes. The hybrid mode is optimized for low-power WSNs since routes are only established on demand. The difference of the hybrid mode to reactive routing strategies is that routing messages are periodically transmitted to maintain already established

routes. However, the protocol stops the transmission of routing messages if no data packets are transmitted for a certain time period in order to minimize the routing overhead and the energy consumption. The proactive mode is designed for high data rate networks which have less energy constraints. In this mode, the protocol periodically transmits routing messages to establish routes in a proactive way even in the absence of data traffic. Thus, nodes in the network can immediately transmit data since the route to the destination is already established.

The performance of the SBR protocol was already evaluated in a large number of simulations and in mobile IEEE 802.11 ad hoc networks where the majority of links have symmetric quality characteristics. However, links in WSNs are often asymmetric in terms of link quality which requires additional mechanisms. The current version of the SBR protocol makes use of passive acknowledgments in order to determine whether a link should be regarded as unidirectional or bidirectional. Thus, the nodes periodically transmit - so called - Short Hello Messages (SHM) which are broadcasted into the two hop neighborhood. The originator of a SHM listens on the radio channel for retransmissions of its SHMs in order to determine the characteristic of the links to its neighbor. A link is marked as unidirectional if none of the last three messages is forwarded by a neighbor node.

The authors of the B.A.T.M.A.N protocol introduced an interesting strategy to estimate the receive, echo, and transmit quality as shown in Figure 1. The link quality can then be

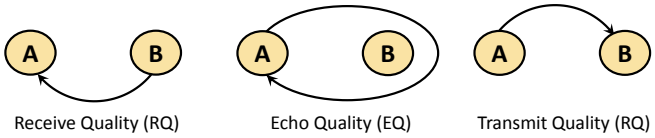


Fig. 1. Link Quality

calculated according to the following equations

$$EQ = RQ \cdot TQ \qquad \Rightarrow \qquad TQ = \frac{EQ}{RQ}.$$

We are currently working on a delay-based approach which defers the forwarding of routing messages depending on the characteristics of the link through which they were received.

V. Medium Access Control

The BPS-MAC protocol uses a new sequential preamble-based medium access strategy which can be adapted to the hardware capabilities of the transceivers. The protocol achieves a very low packet loss rate even in wireless networks with high node density and event-driven traffic without the need for synchronization. This makes the protocol attractive to applications such as structural health monitoring, where event suppression is not an option. Moreover, acknowledgments or complex retransmission strategies become almost unnecessary since the sequential preamble-based contention resolution mechanism minimizes the collision probability. However, packets can still be lost as a consequence of interference or other issues which affect signal propagation.

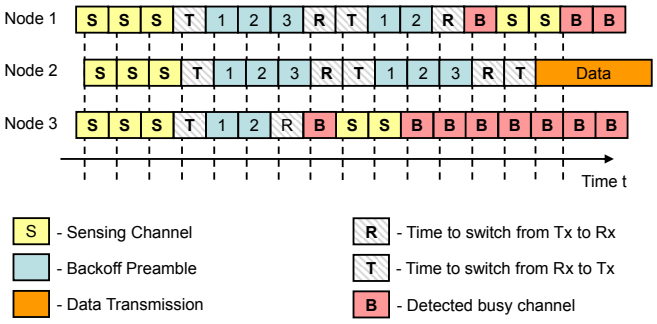


Fig. 2. Example - Sequential Contention Resolution

In the following, we give a brief description of the medium access procedure of the BPS-MAC protocol and discuss its potential in the context of WSNs. Figure 2 shows a typical medium access example for a two-sequence contention resolution. The idea of the BPS-MAC protocol is to use preamble with variable length in order to indicate the intention to access the medium. The protocol applies minimum time units called slots which have to be chosen with respect to the hardware of the sensor nodes. The interested reader is referred - at this point - to [8] which provides a detailed description of the protocol and its mechanisms.

A node - which wants to transmit data - senses the medium for duration of three slots. In the case that the medium is idle, it starts to transmit its preamble which covers the task of a reservation signal. After the transmission of the preamble, it switches its transceiver to receive mode to sense the medium for ongoing transmissions. If the node recognizes a free channel, it switches its transceiver back to tx mode and transmits the next preamble. A node assumes to have access to the medium if it senses a free medium after the transmission of the last preamble.

VI. Security Considerations

Currently the WSN part integrated in AutHoNe is based on a star topology. All sensor nodes are independent of their location and transmit their data to a gateway node. In the future, the star topology should be changed to a multi-hop topology. Depending on the special application within the home network the call for different security levels, such as high or low priority, will occur. Due to the fact that sensor nodes have very limited resources, such as power, memory and computational capacities [12], a simple and source saving technique must be implemented to establish secure communication. Another important fact is the defense against possible attacks against the components in the WSN, which will depend on the application where an attack happens [13], [14].

One possible security mechanism for our application is represented by the SPINS protocol [11], which will be combined with a cluster functionality [15]. More powerful nodes are needed for the cluster head functionality and the gateway node to ensure the connectivity to the Autonomic Home Infrastructure. Those special nodes are responsible for the decoding and

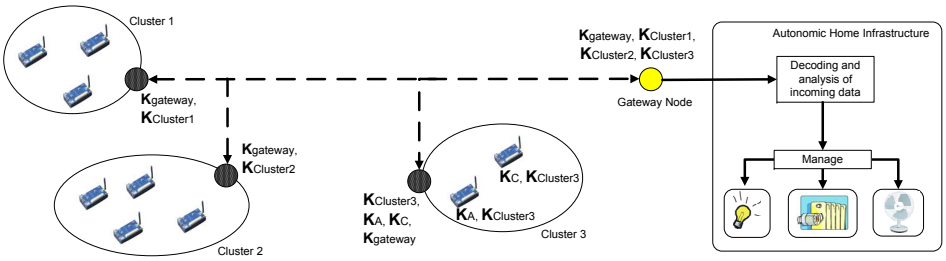


Fig. 3. Secure Communication between Cluster Components and AutHoNe Infrastructure

encoding functionality within the cluster, between the clusters and towards the Autonomic Home Infrastructure.

Figure 3 shows an overview of the given situation. It is assumed that each sensor node in a cluster has a common key pair with its cluster head. Moreover, each cluster head has a common key pair with the gateway node. Consider the following example: Cluster 3 consists of the nodes A, C and its cluster head. For example, node A wants to communicate with node C. They do not have a common key, thus they would waste computational capacities and power for the establishment of a common key, e.g. using Diffie-Hellman. Both nodes have a common key with the cluster head, thus they can use it as an intermediate node. Node A encrypts the messages with the known key $K_{Cluster3}$, transmits the message to the cluster head which re-encrypts the message using the keys K_A and K_C, and transmits the final message to node C. Node C can decrypt the message because it knows the key of the cluster head. As shown in the example, the resources on the individual nodes are saved since the communication and calculations are reduced and less keys must be stored.

As mentioned before, the gateway node is the only node which is able to communicate with each cluster head. Thus, different security levels can be implemented within the established infrastructure. Assume cluster 2 is a cooling room then the temperature should be minus degrees. Thus, cluster 2 has a high priority for cooling which means that the temperature values should be analyzed directly within the Autonomic Home Infrastructure and processed immediately. In this case the common key ensures that the communication between cluster 2 and the gateway node is separated from the communication of the other WSN clusters. The advantage of this approach is represented by the integration of different security levels which can be realized by assigning keys to the corresponding clusters.

VII. Conclusion

In this work, we have introduced the MAUS system which targets the limitations of current wireless sensor solutions for intelligent home automation networks. The introduced solution is based on previous work where we integrated a sensor network in the AutHoNe infrastructure. The MAUS system is currently in a work in progress state since it requires minor changes of existing protocols. However, we are confident that MAUS will achieve a high overall performance due to the fact that it addresses a large number of sensor network related issues which are neglected by the majority of WSN solutions.

References

[1] "Autonomic Home Networking DE Project Page," http://www.authone.de, 2010.

[2] T. Kothmayr, C. Schmitt, L. Braun, and G. Carle, "Gathering Sensor Data in Home Networks with IPFIX," in *Proceeding of the European Conference on Wireless Sensor Networks (EWSN)*, ser. LNCS - Wireless Sensor Networks, T. Voigt, I. Chatzigiannakis, L. Mottola, A. Terzis, A. Krller, N. Tsiftes, T. Baumgartner, and C. Koninis, Eds., vol. 5970/2010, Springer Berlin / Heidelberg. Springer, 2010, pp. 131–146.

[3] C. Schmitt, L. Braun, T. Kothmayr, and G. Carle, "Collecting sensor data using compressed ipfix," in *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN) - Poster Session*, 2010.

[4] A. Klein and P. Tran-Gia, "A statistic-based approach towards routing in mesh networks," in *Proc. IEEE Internatonal Conference on Mobile Adhoc and Sensor Systems MASS 2007*, October 2007, pp. 1–6.

[5] "TinyOS 2.1.1," http://www.tinyos.net/, 2010.

[6] "Berkeley IP Information (BLIP) http://smote.cs.berkeley.edu:8000/tracenv/wiki/blip," 2010.

[7] A. Neumann, C. Aichele, M. Lindner, and S. Wunderlich, "Better Approach To Mobile Ad-hoc Networking (B.A.T.M.A.N.)," Internet-Draft, pp. 1-24, pp. 1–75, April 2008, network Working Group. [Online]. Available: http://tools.ietf.org/html/draft-wunderlich-openmesh-manet-routing-00

[8] A. Klein, J. Klaue, and J. Schalk, "BP-MAC: a high reliable backoff preamble MAC protocol for wireless sensor networks," *Electronic Journal of Structural Engineering (EJSE): Special Issue on Sensor Network for Building Monitoring: From Theory to Real Application*, vol. -, pp. 35–45, December 2009.

[9] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: A short preamble MAC protocol for duty-cycled wireless sensor networks," in *SenSys '06: Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*. New York, NY, USA: ACM, 2006, pp. 307–320.

[10] K. Jamieson, H. Balakrishnan, and Y. Tay, "Sift: A MAC Protocol for Event-Driven Wireless Sensor Networks," in *Proc. of the Third European Workshop on Wireless Sensor Networks (EWSN)*, Zurich, Switzerland, February 2006.

[11] A. Perrig, R. Szewczyk, J. Tygar, V. Wen, and D. Culler, "SPINS: Security Protocols for Sensor Networks," *Wireless Networks*, vol. 8, no. 5, pp. 521–534, 2002.

[12] C. Schmitt and G. Carle, *Handbook of Research on P2P and Grid Systems for Service-Oriented Computing: Models, Methodologies and Applications*. Information Science Publishing, 2010, ch. 46: Applications for Wireless Sensor Networks, pp. 1076–1091.

[13] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *Communications Magazine, IEEE*, vol. 40, no. 8, pp. 102–114, Aug 2002.

[14] J. Walters, Z. Liang, W. Shi, and V. Chaudhary, "Wireless sensor network security: A survey," *Security in distributed, grid, mobile, and pervasive computing*, p. 367, 2007.

[15] N. Vlajic and D. Xia, "Wireless sensor networks: to cluster or not to cluster?" in *WoWMoM 2006. International Symposium on a*, 0-0 2006, pp. 9 pp. –268.

Dynamic Memory Management for Resource Constrained Sensor/Actor Systems

Marcel Baunach
Department of Computer Engineering, University of Würzburg, Germany
baunach@informatik.uni-wuerzburg.de

***Abstract*—Increasing complexity and modularity of today's WSAN applications impose demanding challenges on the system design. This especially affects real-time operation, resource sharing and dynamic memory management. Preemptive task systems are one way to retain good reactivity within dynamic environments. We present our *CoMem* approach for high reactivity and efficient memory usage in such systems. With respect to task priorities and the typically limited resources of sensor nodes, we facilitate compositional software design by providing tasks with runtime information for yet collaborative and self-reflective memory sharing. Thereby, we require no special hardware-support like MMUs but operate entirely software-based.**

I. Introduction

The ever increasing size, pervasiveness and demands on today's wireless sensor/actor networks (WSAN) require modular hardware and software concepts like service oriented abstractions and fine grained code updates. Considering memory management in such systems, we find that current research is still too limited to static concepts. While next generation sensor nodes will more frequently be used as reactive real-time platforms in highly dynamic environments, their true system load varies considerably and can hardly be predicted a priori. Then, preemptive and prioritized tasks are required for fast response on various events but further complicate memory management and reactivity. This is especially true for open systems where real-time and non real-time tasks coexist to reduce hardware overhead, energy issues and deployment effort. In this paper we present our novel CoMem approach for collaborative heap memory sharing and real-time operation within preemptive task systems. CoMem introduces a reflection policy [1] by which programs can become 'self-aware', and may change their behavior according to their own current requirements and the system's demands. In this collaborative manner, CoMem also accounts for task priorities as defined by the developer.

II. Related work and problem definition

Dynamic memory management is subject to intense research efforts [2], [3], [4], [5]. For multitasking systems in particular, heap methods suffer from some inherent flaws, stemming entirely from fragmentation. Thus, a good allocator should

F1 *minimize space* while keeping fragmentation low,
F2 *minimize runtime and overhead* of related functions,
F3 *maximize error detection* or even avoid illegal access,
F4 *maximize tuneability* and support dynamic requirements,
F5 *maximize portability and compatibility,*
F6 *minimize anomalies* to provide good av. case performance,
F7 *maximize locality* by neighboring related blocks, and
F8 *avoid trivializing assumptions* and inadequate constraints.

However, according to [2], any allocator can face situations where it is not optimal or even fails. Especially for systems without MMU or virtual address space, a centralized heap reorganization is hard or even impossible then, since it lacks information about the actual memory usage by the owner tasks.

Thus, the use of dynamic memory is largely avoided for time or safety critical systems [4]. When considering WSAN operating systems, only few support dynamic memory for arbitrary use by application tasks. In particular, no embedded OS provides any mean for reflective dynamic memory (re)organization in case of time-critical sporadic requests or priority inversions when low priority tasks block higher priority tasks by any memory allocation. At most, brute force methods like (energy intensive) swapping, memory revocation or task termination with possibly critical side effects are supported. This is exactly where CoMem applies.

III. The CoMem approach

Reflection based task collaboration is a mighty strategy to share resources on-demand and "upwards" along with the task priorities [6]. We adapt the strengths and benefits for the special case of dynamic memory allocation.

A. Dynamic hints for on-demand resource sharing

Our CoMem approach is generally based on Dynamic Hinting [6], a technique for collaborative sharing of arbitrary resources among prioritized and preemptive tasks. Dynamic hinting analyzes emerging task/resource conflicts at runtime and provides spurious tasks with information about how they can help to improve the reactivity and progress of more relevant tasks. In combination with the basic Priority Inheritance Protocol (PIP) [7], it improves and stabilizes the overall system performance. According to PIP, a task t's priority is raised iff it blocks at least one other task with truly higher priority by means of at least one so called *critical resource*. Only then, dynamic hinting immediately passes a hint to t indicating this priority inversion and 'asks' for releasing at least one critical resource quickly. While this facilitates the on-demand release and handover of blocked resources, passing such hints is not trivial in preemptive systems, since from the blocker's view, these occur quasi-asynchronously and regardless of its current task state. Hence, we'll consider two options:

- Early Wakeup: When in *waiting* state (i.e. suspended by a blocking function), t will immediately be resumed and the return value signals this special situation.
- Hint Handler: When in *ready* state (i.e. preempted by another task) a hint handler is injected into t's execution and operates resources in a quasi-preemptive way.

Anyway, hints are passed instantly and only when blocking really occurs. Then, the task decides between following or ignoring them. When following, it saves the resource state before releasing it. This will immediately cause an implicit task self-preemption due to the resource handover. On resumption it re-allocates the resource and restores its previous state.

B. CoMem for dynamic memory allocation

Since memory is commonly a scarce resource for sensor nodes, it needs to be shared among tasks to achieve a higher integration density for future, versatile systems and applications. As many tasks run rather seldom, a static allocation would leave valuable space unused for long periods. Yet, even rarely running tasks might be subject to tight timing constraints and request memory only upon certain events (e.g. triggered by environmental interactions, see F4 and Section V).

To reduce overhead in terms of task count, context switches, stack space, and to avoid indirect priority inversion by low priority tasks calling high priority memory management functions, we execute the memory management functions entirely within the context of the calling tasks (→F2, F4). For temporally limited blocking, we extended our `malloc()` function by a timeout parameter and provide the memory management subsystem with information about how long we are willing to wait in worst case. Simultaneously we supply a defined amount of time for reorganization of the heap space. Finally, two elementary options exist for this step:

- release memory blocks (e.g. dismiss or swap data first)
- relocate memory blocks (e.g. for compaction)

Please recall, that revoking or moving memory *without* signaling this to the owner task is complicated or even impossible in most cases. Not even data structures which are just accessed relative to the block base addresses (like stacks) can simply be relocated: expired addresses might still reside in registers or CPU stages, then. Much worse, affected peripherals like e.g. DMA controllers cannot be updated automatically and would still transfer data from/to old addresses. In such situations not even task termination and restart is a valid solution. Instead, this can only be handled by the owner task which has complete knowledge about the memory usage and *all* dependencies. Thus, the central idea is to inform those tasks which cause the denial of memory for higher prioritized tasks (→F4). We also advise them whether releasing or relocating their blocks would solve this problem most suitably. Finally, this triggers a self-controlled but on-demand heap reorganization by means of some helper functions like relocate or release (see Fig. 1).

IV. CoMem implementation and usage

The basic idea behind CoMem might be applied as integral concept for many (embedded) operating systems if these support preemptive and prioritized tasks plus a timing concept that allows temporally limited resource requests. We extended *Smart*OS [8] since it fulfills these requirements, offers quite common characteristics, and thus is a good representative for the adaptation of similar systems (→F5). While *Smart*OS supports PIP, tasks may hold several resources simultaneously. Allocation and deallocation orders are arbitrary and independent. Apart from the CPU, resources are always treated as non-preemptive and will never be withdrawn.

A. CoMem implementation details

Regarding the tight performance and memory constraints of many embedded systems, CoMem is limited to three central functions and one Control Block (MCB) for each allocation:

```
1 typedef struct {   // The Memory Control Block (MCB)
    unsigned int size;   //1W: size in machine words
3   volatile int *base;  //1W: block start address
    Resource_t   broker; //2W: associated OS resource
5   MCB_t        *next;  //1W: linked list pointer
  } MCB_t;               // Total RAM size: 5W
```

Since we want tasks be be informed *immediately* if they block a higher priority task due to a dynamic memory allocation, we associate one *Smart*OS resource – a so called *broker resource* – with each allocated memory block. Thus,

1. we implicitly adapt the underlying resource management policy (e.g. PIP) for the memory management: All system resources and memory blocks are treated in the same way, and respect the task priorities equally.
2. CoMem can be implemented as library and does not produce additional overhead within the kernel.

Let's take a look at the function interactions as depicted in Figure 1. Internally, **`malloc(...)`** loops until a request succeeds or its timeout is reached (Line L4): Initially each retry attempts to find sufficient free space on the heap (first-fit, L5). On success (L7), the corresponding broker-resource b_m is locked by the caller and we are done. Since b_m belongs to the block owner $\sigma(m)$ then, it is sufficient for another task with higher priority to request this very resource if it is blocked by $\sigma(m)$. Indeed, this is exactly what happens if sufficient space is not available but a disturbing memory block m' was found (L11). By the resource request (L12), PIP adapts the active priority $p(\sigma(m'))$ of the blocking owner $\sigma(m')$. If dynamic hinting is enabled, the resource manager immediately passes a hint to $\sigma(m')$ to indicate its disturbing influence. If $\sigma(m')$ reacts by releasing or relocating its block m' before the timeout τ has expired, it also releases m'_b (`free()`:L3, `relocate()`:L5) to indicate the changed memory situation and to trigger a new retry for m. If no spurious task/block was found (L14), `malloc()` waits for the next modification to the heap space. Again, one more retry is triggered if there is still some time left. If the timeout has expired, `malloc()` stops and returns 0 (L18).

The remaining problem is how to *reasonably* select a blocking MCB m' for generating a hint on. While scanning for free space (L5), we search for two types of MCBs: The first would provide the requested space if it was relocated and the

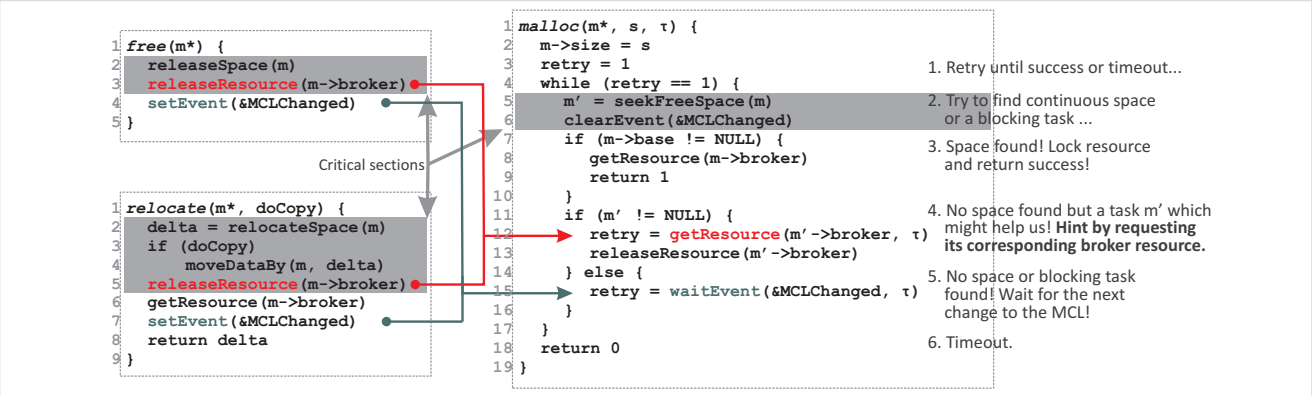


Figure 1. CoMem function interactions

other one if it would be released entirely. While the first takes precedence, the corresponding block with the lowest priority owner is selected for hinting. Thus, along with the hint we also pass the advice for a suitable reaction.

Finally, `free()` and `relocate()` are rather simple:
`free(m)` simply removes the specified MCB m from the memory and releases the broker resource b_m. Finally, it triggers a special event to indicate the heap update.
`relocate(m)` seeks a new location for the specified block m (cyclic next-fit) by which more continuous free space becomes available (L2). If requested, it also moves the data. Then, it temporarily releases its own broker-resource b_m (L5/6) and also triggers the update event (L7) to resume waiting tasks.

V. Test bed

For our test bed, we used *Smart*OS for the Texas Instrument's MSP430 family of microprocessors, since these are found on a large variety of sensor nodes. Requiring 5 kB of ROM and 60 B of RAM for the whole OS kernel and the CoMem library, the typically small memory of sensor nodes was considered carefully to leave sufficient space for the actual application.

A. Dynamic memory stresstest

This scenario analyzes our approach under extreme conditions with n tasks $t_0, \ldots, t_{n-1}$ and many concurrent memory requests. We used ascending base priorities $P_{t_i} = i$ and each task executed the same code repeatedly: (1) sleep, (2) request memory, (3) operate on the memory, (4) release the memory. The duration of step (1), the CPU time for step (3) and the size of the requested memory blocks were randomized for each iteration. This way, we obtained significant heap space fragmentation and task blocking which needed handling at runtime. Though we specified infinite timeouts τ, each task measured the execution time δ of `malloc()` and logged its min., max. and av. allocation delays $\delta_{min}, \delta_{max}, \delta_{av}$. It also registered the number of received hints. For comparing the allocation delays in relation to the task priorities, we applied two non-collaborative and two collaborative policies P1-P4:

- P1 Classic: We omitted the request for a blocking task's broker resource during `malloc()` (L12). Instead we always waited for heap modifications (L15) if no continuous space was found. This avoided PIP, hints and the chance for collaborative memory sharing entirely.
- P2 PIP only: We implemented `malloc()` as described but simply ignored the emerging hints. Though a blocking task could not collaborate then, PIP raised its active priority to the task it blocked and it received CPU time quickly.
- P3 Hint Handlers: Each task supplied a hint handler for immediate injection into its own execution flow when blocking a higher prioritized task. This simulated blocking while in ready or preempted state.
- P4 Early Wakeup: Finally, the tasks did sleep while holding a memory block. Yet, they were resumed immediately when blocking a higher prioritized task. This simulated blocking while in waiting or suspended state.

For collaboration under P3 and P4, each task treated its hints as described in Section III-A.

We configured the test bed using several configurations, but just present the analysis for $n = 10$ tasks, block sizes $s_B \in \{32, 64\}$ words and heap sizes $s_H \in \{320, 480, 640\}$ words.

As expected, all allocations succeeded immediately when sufficient heap space $s_H = 640$ words was available to serve all requests even under worst case fragmentation. We did this cross-check to obtain a reference time δ_{ref} for other heap sizes. Indeed, the average allocation delay settled around $\delta_{av} = \delta_{ref} = 280$ µs for each task and policy.

Selecting $s_H := 10 \cdot \frac{32+64}{2} = 480$ words (the required heap size for the average case) already shows the benefits of our approach (→Fig 2a). While the non-collaborative policies deliver almost uniform average allocation delays around 161 ms (P1) and 69 ms (P2), both do not reflect the tasks' intended base priorities at all. Using hints manages to reliably signal tasks about their spurious influence and allows them to react adequately. Considering the average *and* maximal allocation delays, the task priorities are visibly reflected by both collaborative policies P3 and P4. Compared to P1 and P2, not even the lowest priority task t_0 suffers from significantly increased delays, while several high priority tasks are still very

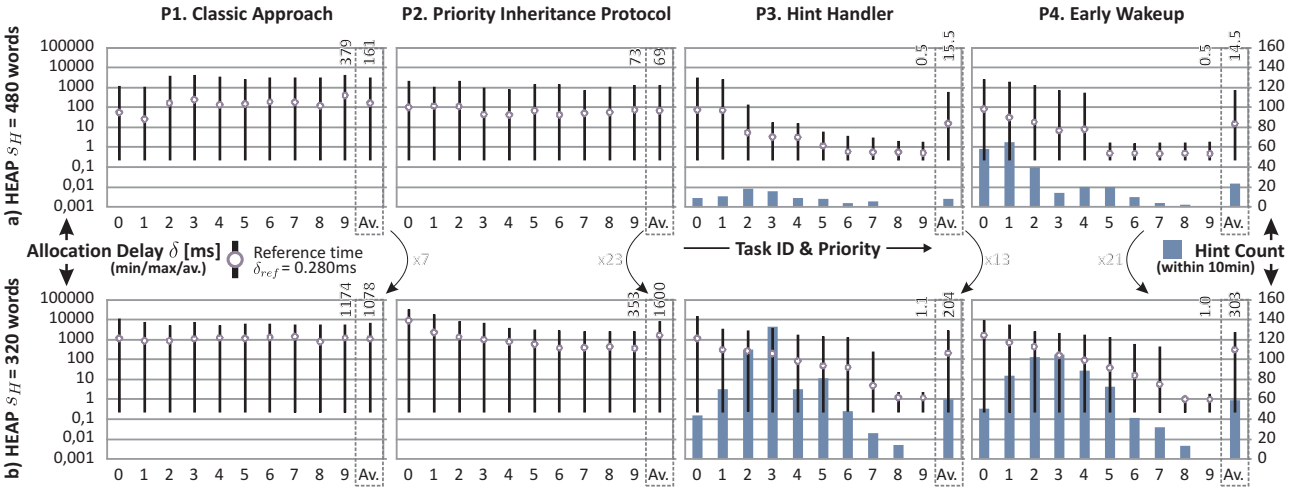


Figure 2. CoMem stresstest results for different heap sizes (ascending base priorities for $n = 10$ tasks)

close to the reference time $\delta_{ref} = 280$ µs.

Reducing $s_H := 10 \cdot 32 = 320$ words increases competition and allocation delays to be even more demanding (→Fig 2b). Still, the different task priorities are not visible for P1 and weakly visible for P2. However, their average allocation delay increased by factor 7 and even 23. Since blocking occurs more often now, the hint count also increases for the collaborative policies. Yet, these still manage to serve tasks according to their intended relevance: the two most important ones still achieve an average delay of $\delta_{av} \approx 1$ ms while the lowest prioritized ones are still at least as reactive as with the non-collaborative approaches. Similar results are visible for δ_{max}.

This testbed addressed allocation delays for dynamic memory in case of sporadic requests and varying task priorities. We pointed out that dynamic dependencies (via broker resources) between blocking and blocked tasks can already reduce these delays in general, and account for the specific task priorities in particular. While PIP already showed rudimentary success for heavy load situations, hints boosted this effect significantly.

VI. Conclusion and Outlook

In this paper, we outlined our CoMem approach for collaborative memory sharing among preemptive tasks in reactive systems. We showed, that CoMem can help to improve and stabilize the overall system performance by optimizing memory allocation delays. Apart from F3 (protection) and F7 (locality) it also considers the feature requests from Section II. In particular, individual task base priorities are considered carefully to keep each task's progress and reactivity close to its intended relevance. By providing spurious tasks with information about how to reliably reduce the blocking of more relevant tasks, the hints allow them to collaborate implicitly and without explicit knowledge of each other. This even reduces priority inversions and achieves memory allocation delays which are mainly limited by the pure handover overhead.

As a reflective concept, CoMem allows individual tasks to decide dynamically between collaborative or egoistic behavior with respect to their current conditions and other tasks' requirements. Thus, we initially can not guarantee any time allocation limits since these highly depend on the behavior of the blocking tasks. In addition, unless a MMU is available, our concept cannot protect memory against unauthorized access but only coordinate its exclusive sharing (→F3).

The test bed showed, that the effective use of prioritized tasks for creating reactive open systems is even feasible on small embedded devices like sensor nodes: High priority tasks almost achieved the theoretical best case reactivity while low priority tasks did hardly lose performance. Even if used sparsely, our approach always proved to be better compared to non-collaborative operation. Though a well-thought application design still remains elementary, compositional software is already facilitated. In general, our approach is not necessarily limited to sensor/actor networking but may also extend other embedded systems.

References

[1] N. Audsley, R. Gao, A. Patil, and P. Usher, "Efficient OS Resource Management for Distributed Embedded Real-Time Systems," in *Operating Systems Platforms for Embedded Real-Time applications*, 2006.

[2] P. R. Wilson, M. S. Johnstone, M. Neely, and D. Boles, "Dynamic storage allocation: A survey and critical review." Springer-Verlag, 1995.

[3] H. Min, S. Yi, Y. Cho, and J. Hong, "An efficient dynamic memory allocator for sensor operating systems," in *SAC '07: ACM symposium on Applied computing*, 2007.

[4] M. Masmano, I. Ripoll, P. Balbastre, and A. Crespo, "A constant-time dynamic storage allocator for real-time systems," *Real-Time Syst.*, vol. 40, no. 2, pp. 149–179, 2008.

[5] G. Teng, K. Zheng, and W. Dong, "SDMA: A simulation-driven dynamic memory allocator for wireless sensor networks," *Int'l Conference on Sensor Technologies and Applications*, 2008.

[6] M. Baunach, "Dynamic hinting: Real-time resource management in wireless sensor/actor networks," in *15th IEEE Int'l Conference on Embedded and Real-Time Computing Systems and Applications*, 2009.

[7] L. Sha, R. Rajkumar, and J. P. Lehoczky, "Priority Inheritance Protocols: An Approach to Real-Time Synchronization," *IEEE Trans. Comput.*, vol. 39, no. 9, pp. 1175–1185, 1990.

[8] M. Baunach, R. Kolla, and C. Mühlberger, "Introduction to a Small Modular Adept Real-Time Operating System," in *6. Fachgespräch Sensornetzwerke*. RWTH Aachen University, 16.–17. Jul. 2007.

Smart Energy Module for Wireless Sensor Nodes

Juergen Jessen, Marcus Venzke, and Volker Turau
Institute of Telematics
Hamburg University of Technology
Hamburg, Germany
Email: {juergen.jessen, venzke, turau}@tu-harburg.de

***Abstract*—This paper presents the design of a universal energy module for wireless sensor nodes, which supports a wide range of energy harvesters and energy storages. The focus is on the efficient conversion and storage of energy and to provide hardware support for higher level management functions. The module facilitates maximum power point tracking to improve the harvester efficiency. By supporting hybrid harvesters, the module can make better use of the time-dependent availability of environmental energy sources.**

I. Introduction

Continuous energy supply is a common problem for wireless sensor nodes. Energy harvesting is a suitable solution for a potential unlimited lifetime of wireless sensor nodes. Energy harvesters gather electric energy from different environmental energy sources, which are rarely steady. The energy needs to be buffered for periods of no harvested energy and for current peaks, which occur from typical node duty-cycling.

In practice an unlimited lifetime is impossible because of aging effects of the components. Especially the harvester and the rechargeable energy buffers degrade and can supply or store less energy over the years. The choice of the components is a trade-off between lifetime, price, and size. Providing power to the node efficiently is a great challenge, depending on the choice of components, node hardware, and the application. The most simple design by directly attaching a harvester to the energy storage and node is only efficient for carefully adapted components and if environmental conditions do not change. This limits the developer's choices of the harvester and energy storage. The system's reliability can be improved by supporting a second harvester. A single harvester like photovoltaic cells produce energy very unreliable for extended periods (as in winter). A complementary harvester, like a wind turbine, can solve this issue. The ability for supporting a wide range of harvesters allows mass production and therefore cheap modules. It reduces the non-recurring engineering costs for the development of an energy module and lets the developer focus on his main topic, the wireless sensor node.

This paper presents the design for a universal energy module, which supports more than one harvester connected at the same time (hybrid harvester). The harvested energy is efficiently converted to the needs of the consumer, using maximum power point tracking to further improve efficiency. The energy is stored in a multi-staged energy storage, which is designed to minimize aging effects and improve reliability. Besides proper and fault-proof operation the circuit must also provide data about the module state, which can be used for energy management on higher software levels. This universality comes at the cost of a higher system complexity and possibly lower system-efficiency compared to specially designed fixed size and type harvester energy modules. Handling this complexity, discussing the problems, and giving possible solutions are the topics of this paper. The results are currently validated in a prototype implementation.

II. Related Work

Energy harvesters are used in many projects. Especially photovoltaic cells are commonly used, because they are cheap, easy to use and efficient. The energy module design described in this paper is an evolved prototype for the IRIS platform [1], which was presented in [2] and [3].

A similar approach is presented in Enviromote [4]. Their goal is a cheap and easy circuit design for harvesting solar energy and storing it in a rechargeable NiMH accumulator. The concept of a multi-stage energy storage is shown in the Prometheus project [5]. A supercapacitor (electric double-layer capacitor, supercap) is used as a primary buffer, while a rechargeable lithium-ion battery prolongs the node lifetime for extended periods of little or no sunshine.

In order to maximize the efficiency of energy harvesting, low power consumption of the components is necessary. The second influence factor is the efficiency of voltage conversion between the harvester, the energy storage, and the wireless sensor node. A good example is shown in [6] for the IRIS platform [1]. This design stores the energy in lead acid batteries and operates the photovoltaic cells at a static maximum power point to further improve efficiency.

Efficient circuits for different harvesters including a minimized version of maximum power point tracking (MPPT) are presented in [7]. The focus is on self-powering and a maximized power output even for sub mW harvesters. A complete MPPT system is shown in the Everlast platform [8]. Here the solar energy is stored in a supercap only, which allows very long operation times of up to 20 years, but has little reserves for extended periods of no sunshine.

The mentioned systems focus on a single harvester only, which is limited to solar energy in most cases. A universal approach is not intended, rather the optimization for a specific task. A universal energy module can support more types of harvesters and even hybrid harvesters with a single module and combines the advantages of the systems mentioned. It

should automatically adapt to a wide range of harvesters and energy storages, using efficient energy conversion and MPPT to further increase the efficiency.

III. Universal Energy Module Architecture

A universal energy module is the link between energy *harvester* and wireless sensor *node*, presented as flow of energy through the module in Fig. 1. It transparently provides the capability of storing and efficiently converting energy, independent of the used harvester and node. The main control functions are combined in the middle *control block*. Some of the components are optional (light colored). The energy from a *selected* harvester is *rectified*, and *MPP tracking* is used to improve efficiency. Depending on the charge strategy, an *energy storage* is *selected* and charged using *voltage regulation* and *charge control*. The most appropriate energy storage powers the node at optimal voltage controlled by the *regulator*.

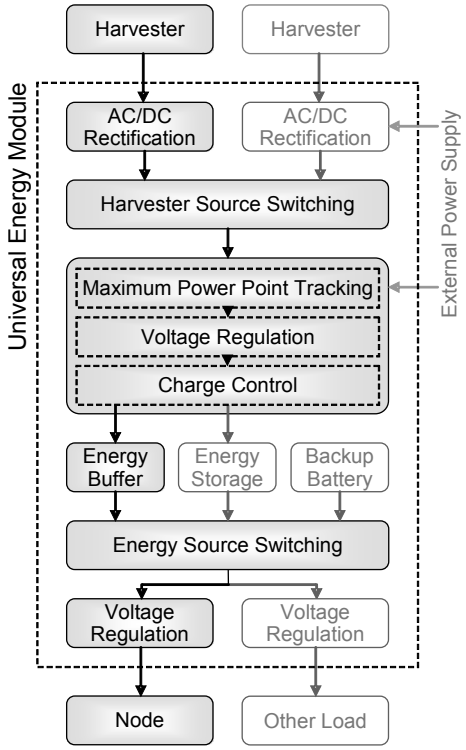


Fig. 1: Universal Energy Module

IV. Energy Harvester

While photovoltaic is the most common harvester type, other harvesters types should be supported by a universal energy module as well, such as thermoelectric, piezoelectric, electromagnetic, electrostatic, and RF antenna. The hybrid harvester concept improves reliability of the system by increasing usability of environmental energy sources. Instead of increasing the peak energy production by using both harvesters at the same time, the steady energy flow is improved by selecting the most productive one, resulting in less necessary energy storage capacity and less hardware effort.

Each harvester shows a different voltage-current characteristic and has another optimal working point, which additionally dependents on the environmental conditions. Another challenge is to make use of power, where voltage and current are phase-shifted, for example piezoelectric harvesters. To counter the effect of a parasitic capacitance in the harvester, an actively controlled inductance is required in parallel to the harvester.

Most piezoelectric and electromagnetic harvesters produce an AC current. The current must be rectified to DC current first and the output signal smoothed by a simple low-pass filter. The AC/DC rectification can also be used as reverse voltage protection for the harvester or to produce energy when the polarity of the harvester reverses. This is for example useful for thermoelectric harvesters, when the direction of the heat flow becomes reversed. Instead of using a full-wave bridge rectifier, a Delon or Greinacher circuit allows doubling the output voltage (cascaded even higher multiplier). This is especially useful, if the harvester output voltage is low and needs to be boosted anyway. However, each diode stage decreases efficiency, even when using low forward-voltage drop Schottky diodes.

To further increase the efficiency of the rectification, a synchronous active rectifier can be built by replacing the diodes of the full-wave bridge rectifier with MOSFETs. The necessary control circuit can be built from a simple analog comparator comparing the input AC voltage levels and switching the correct path in the rectifier. To ensure proper startup when no energy reserves are available for the control circuit, the diodes remain in the system, but are bypassed by the MOSFETs once the control circuit is running. This design needs a higher input voltage for start-up, but once the system is running the voltage drop is lowered and the efficiency increased. The active rectifier is also useful for DC harvesters to prevent any current flow into the harvester, when it is not providing enough power. The usually used passive diode can be omitted, increasing the efficiency of the circuit, because in this case MOSFETs cause less power loss than a diode.

V. Energy Storage System

Using more than one energy storage is useful, because each technology has its advantages and disadvantages. A multi-staged design combines several storage technologies, altering the combination and capacity to optimize the storage system for a specific application. Not all stages are needed in any case. A self-powered control circuit ensures that energy for the wireless sensor node is only supplied from one storage stage at the same time, switching the energy storages in the presented order. The typical requirements for wireless sensor nodes are a high energy density, a high number of recharge-cycles, a long lifetime durability, a high temperature stability, and a small size and price. A suitable design for an energy module consists of three stages, a short-time energy buffer, a long-time energy storage, and a backup battery.

An energy buffer is used to store small amounts of energy to buffer the typical periodic power cycle of the harvester. To reduce wear of the other storage systems, whenever possible, the energy buffer stores the harvested energy and supplies the consumer. Supercaps are ideal as primary energy buffer [9].

A long-time energy storage can ensure proper operation for long periods, if the harvested energy is not sufficient. The storage is rechargeable, but the recharges are allowed to degrade the energy storage and have less efficiency. Therefore the use of the secondary stage should be minimized, favoring the energy buffer. A suitable energy storage is characterized by a high energy density and a low self-discharge. Good examples are Li-ion, low self-discharge NiMH accumulators, or second generation lithium accumulators like LiFe-PO_4.

A non rechargeable battery can act as a backup, if the first systems should fail. Lithium batteries are most suited for this purpose. They have very high energy densities and show a very low self-discharge of only 1% per year.

VI. Energy Module

An energy module is developed to meet the mentioned requirements. It operates as a stand-alone system and can therefore supply any typical wireless sensor node. This requires a dedicated microcontroller on the energy module, because analog circuitry would be too complex to fulfill all tasks. The microcontroller is the major consumer of energy on the energy module. To reduce power consumption the low-power microcontroller Atmel ATtiny88 has been selected [10]. The microcontroller is clocked by a built-in low frequency RC-oscillator. An accurate clock frequency is not necessary, therefore the advantages of low power consumption and fast start-up of the RC-oscillator outbalance any other clock source. To further decrease the necessary energy, the microcontroller is operated in duty-cycle mode, sleeping most of the time.

Some parts of the energy module are not controlled by the microcontroller, for example the AC/DC rectification circuit, parts of the MPPT, and the energy storage selection. Although this would be possible, it is more energy efficient to use an analog circuit, because it would need too many resources of the microcontroller, preventing the use of sleep modes. The control of the input power path of the universal energy module is shown in the system block diagram in Fig. 2. To simplify the architecture, only one harvester and energy storage are shown and the output voltage generation is left out.

The prototype energy module is built to support input voltages of 0.7-5.5 V at a current range of 100 μA-200 mA. The module supports active AC/DC rectification and maximum power point tracking. For energy storage a supercap, a LiFe-PO_4, and a lithium battery have been selected.

A. Efficient Energy Conversion

Wireless sensor nodes need a constant input voltage, typically in the range of 1.8-3.3 V. The energy storage rarely has the same voltage. For example when Greencaps [9] are used, the supercap voltage will range form 0 V to 2.7 V. A step-up (boost) DC/DC converter is used to increase the input

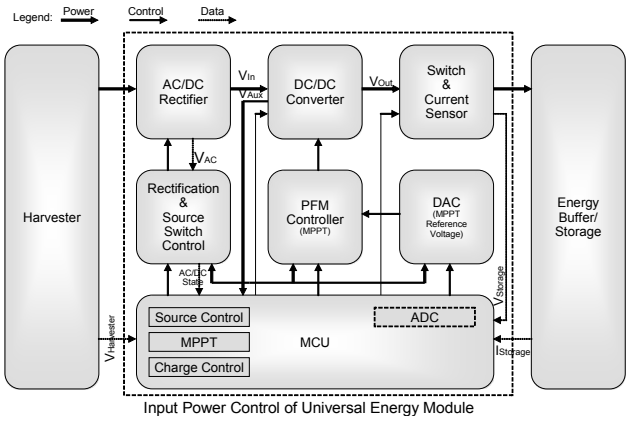


Fig. 2: System Block Diagram Input Power Control

voltage. In case of a higher storage voltage, instead of a low-dropout regulator a step-down (buck) DC/DC converter is usually more efficient. The best solution is a single buck-boost DC/DC converter, enabling up- and down-voltage conversion to efficiently operate the circuit independent of the energy storage voltages. The voltage regulator can be adjusted to operate at the lowest possible voltage to reduce the power for the microcontroller and transceiver. Optionally the voltage is adjustable by higher level energy management software. This would enable dynamic voltage scaling which can be used in combination with frequency scaling to decrease the energy consumption of the system. Additionally a secondary software-adjustable voltage regulator is useful to operate a sensor or actuator needing a different voltage than the node.

To maximize energy storage efficiency, mainly the first stage energy buffer is used. For supercaps the efficiency for storing energy is nearly 100%, because of the low equivalent series resistance (ESR). Li-ion based technology still offers a high charge-discharge total energy efficiency of 90%, while for example NiMH offers only 70%. The efficient conversion of voltages is done by dedicated DC/DC converters. Generally the efficiency of conversion is higher, when input and output voltage differ less. The conversion of the harvester input voltage to the energy storage voltage is done by a DC/DC converter, which is controlled by the MPPT (Section VI-D). The output voltage is provided by the DC/DC converter TPS61220 from Texas Instruments [11]. The converter can operate at an input voltage down to 0.5 V, allowing to make use of more than 95% of the energy stored in a 2.7 V supercap.

B. Energy Monitoring and Management

The energy module periodically monitors the system-state. Primarily this data is needed for MPPT, charge-control, and energy management. Measurements are realized by the built-in analog-to-digital-converter (ADC) of the microcontroller and a reference voltage. The critical controls like AC/DC rectification and parts of MPPT are done in dedicated hardware, therefore fast measurements are not necessary and the intervals between measurements can be extended to reduce energy consumption. The following parameters are monitored:

- Primary and secondary harvester voltage
- AC/DC controller state (digital input)
- Harvester-sided DC/DC converter output current
- Energy buffer, storage, and backup battery voltage
- Board temperature (μC on-chip sensor [10])

The microcontroller on the energy board additionally acts as a platform for higher level energy management functions. The ability of controlling all external circuits, input and output currents allows numerous options for energy management functions such as estimation of remaining energy, prediction of remaining runtime [2], or self-calibration of the capacity of connected energy storages [3].

C. Charge Control and Fail-Proof Operation

The energy module is built to support different accumulator technologies. The charging parameters depend on the used accumulator technology and are preconfigured in the microcontroller internal EEPROM.

The most important part while charging is monitoring the maximum energy storage voltage. Timing requirements and maximum current are of minor importance, because the harvester maximum current is usually below the energy storage maximum charging current. Nonetheless the charging current can be measured (harvester-side DC/DC output current) and be controlled adjusting the MPPT control parameters (Section VI-D), overriding the AC/DC rectification controls, or duty cycling the connection to the energy storage. Additionally a deep discharge must be prevented.

The safety features in software prevent hardware damage and ensure a stable system. Another aspect is the bootstrap capability of the energy module. Even if all energy storages are discharged, the system can regenerate. When the harvester can gather enough energy, the active AC/DC rectification is bypassed by the diodes. This will power the DC/DC converter. At this time the energy storages are still disconnected, but the controller can start up. When the controller becomes active, it can enable active AC/DC rectification and MPPT to increase the efficiency of energy harvesting. At that point charging is started and when the energy buffer reaches a sufficient energy level, the consumer can be activated again.

D. Maximum Power Point Tracking

Each harvester has an optimal working point, where the harvested power is maximized, the maximum power point (MPP). This point is not fixed, but depends on the amount of harvestable environmental energy, the temperature, and the wear of the harvester. To produce the maximum energy in all situations, the point of operation needs continuous adjustment, which is called maximum power point tracking (MPPT).

The underlying problem is that internal resistances are not matched. Adding a DC/DC converter in between, the internal resistance becomes controllable by adjusting the duty cycle. A boosting DC/DC converter furthermore allows the use of harvesters, whose output voltage is less than the energy storage voltage. The low ESR of energy storages like supercaps allows only short pulses for charging, because otherwise the harvester voltage would drop too much. Therefore the duty cycle is not controlled by a pulse width modulation (PWM), but a pulse frequency modulation (PFM).

The system block diagram in Fig. 2 shows the control dependencies for MPPT. The PFM signal is generated from an analog comparator, which enables the DC/DC converter whenever the harvester voltage is higher than a reference voltage. Adding hysteresis stabilizes the comparator. The reference voltage is generated from the microcontroller using a digital-to-analog converter (DAC). The DAC is configured by the microcontroller, to follow the MPP of the harvester. It can be adjusted gradually, by implementing a hill climbing algorithm, using the harvester power as weighting function.

VII. Conclusion and Next Steps

Many issues have to be considered to develop a universal energy module for wireless sensor nodes. They range from the efficient use of possibly different harvesters and the preservative application of suitable storage technologies to efficiency, fault-proof, monitoring and energy management topics. The presented energy module design solves these issues as a standalone system with a dedicated microcontroller.

Future work involves improving the efficiency and implementing more parts in analog circuitry, lowering the absolute power necessary to operate the energy module. The energy management functionality is evolving, automatizing parts of the configuration and integrating high level energy management functions from the wireless sensor node to the energy module, allowing other developers to focus on wireless sensor network functionality rather than energy problems.

References

[1] Crossbow, *IRIS Wireless Measurement System - Datasheet*. [Online]. Available: http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/IRIS_Datasheet.pdf

[2] C. Renner, J. Jessen, and V. Turau, "Lifetime Prediction for Supercapacitor-powered Wireless Sensor Nodes," in *Proc. of the 8th GI/ITG KuVS Fachgesprch "Drahtlose Sensornetze" (FGSN'09)*, August 2009, pp. 55–58.

[3] C. Renner and V. Turau, "CapLibrate: Self-Calibration of an Energy Harvesting Power Supply with Supercapacitors," in *Proc. of the GI/ITG Workshop on Energy-aware Systems and Methods*, February 2010.

[4] V. Kyriatzis, N. S. Samaras, P. Stavroulakis, H. Takruri-Rizk, and S. Tzortzios, "Enviromote: A New Solar-Harvesting Platform Prototype for Wireless Sensor Networks / Work-in-Progress Report," in *Proc. of the Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '07)*, Athens, Greece, 2007.

[5] X. Jiang, J. Polastre, and D. Culler, "Perpetual Environmentally Powered Sensor Networks," in *Proc. of the Intl. Symposium on Information Processing in Sensor Networks (IPSN '05)*, Los Angeles, USA, 2005.

[6] C. Lange, "Energiegewinnung für drahtlose Sensorknoten (Diploma Thesis)," Master's thesis, Hamburg University of Technology, Oct. 2008.

[7] P. Spies, "Adaptive Power Management Circuits for Selfpowered Systems," in *Presentation on Energy Harvesting and Storage Europe 2010*, Munich, Germany, May 2010.

[8] F. Simjee and P. H. Chou, "Everlast: Long-Life, Supercapacitor-Operated Wireless Sensor Node," in *Proc. of the Intl. Symposium on Low Power Electronics and Design (ISLPED '06)*, Tegernsee, Germany, 2006.

[9] SAMWHA, *Green-Cap EDLC*. [Online]. Available: http://www.samwha.com/electric/templatedirs/guest/list_pdf1/DE.pdf

[10] Atmel, *Datasheet ATtiny48/88*, June 2010. [Online]. Available: http://www.atmel.com/dyn/resources/prod_documents/doc8008.pdf

[11] T. Instruments, *Datasheet TPS 61220*, Jan. 2009. [Online]. Available: http://focus.ti.com/lit/ds/symlink/tps61220.pdf

Benchmarking of WSN Solutions and IEEE 802.15.4-2006 PSSS based Solutions

Andreas C. Wolf
Managing Director
Dr. Wolf Wireless GmbH
Teltow, Germany
aw@dw-w.com, www.dw-w.com

Matthias Mahlig
R&D
IHP GmbH
Frnkfurt Oder, Germany
mahlig@ihp-microelectronics.com,
www.ihp-microelectronics.com

Abstract—**PSSS (Parallel Sequence Spread Spectrum) [1] technology is the basis for the PHY of the new IEEE802.15.4-2006 standard with the enhancement of the data rate from 20 kbps to 250 kbps for the European area. Robustness against multipath fading and interference is also enhanced and makes the sub 1 GHz PHY highly attractive. Compared to 2.4GHz solutions there is lower attenuation in the transmission path.**

I. Introduction

The sub 1 GHz PHYs of the IEEE 802.15.4-2003 standard offer only 20 kbps for Europe/ETSI (European Telecommunications Standards Institute) and 40 kbps for the FCC region. Compared to the 250 kbps possible at 2.4GHz, the data rate was unattractive, especially for WSN (Wireless Sensor Networks) with many nodes. For the ETSI region it has to be taken into account that there is a duty cycle limitation of 1%. That causes average data rates of not more than 200 bps for the IEEE 802.15.4-2003 PHY. The peak data rate for the sub 1 GHz IEEE820.15.4-2006 PHYs (ETIS/FCC) is 250 kbps as with the 2.4 GHz PHY.

The coverage is for sub 1GHz bands better than for the 2.4 GHz band. Simulations with a ray tracing tool underline this fact. Figure 1 shows the received power for a 2.4 GHz transmission for a LOS (Line of Sight) and a NLOS (No Line of Sight) area.

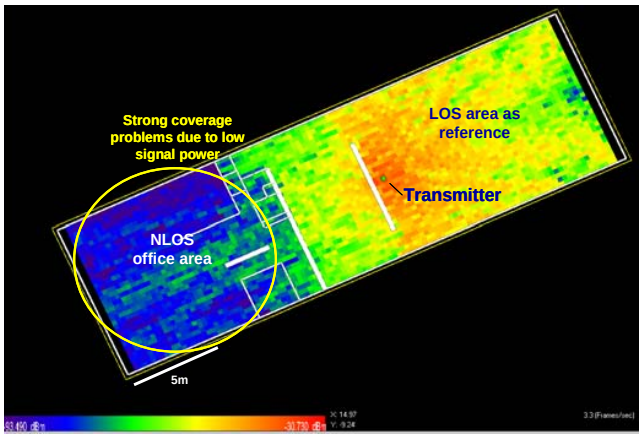


Figure 1. Coverage at 2.4 GHz in LOS and NLOS areas. Received power: blue -93,5 dBm, red -30 dBm

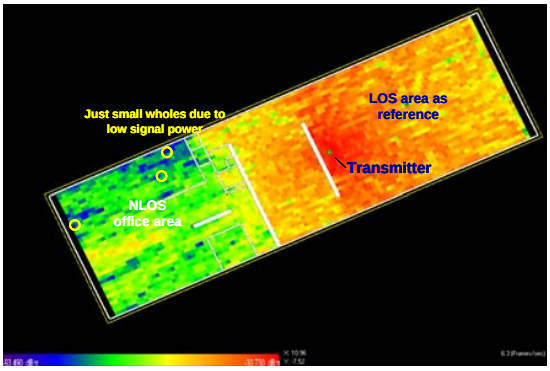


Figure 2. Coverage at 868 MHz in LOS and NLOS areas. Received power: blue -93,5 dBm, red -30 dBm

The 868 MHz example in figure 2 shows that the received power is significantly higher. Also the expected interference is in the sub 1 GHz better than in the 2.4 GHz band, because WLAN and Bluetooth are occupying the 2.4 GHz band.

The motivation for enhancing the data rate for the sub 1 GHz PHY in the IEEE 802.15.4-2006 standard was to combine the attractive coverage of the sub 1 GHz band and the low interference with the high data rate of the 2.4 GHz PHY. Especially for the ETSI area with the 1 % duty cycle limitation the increased data rate was necessary.

II. PSSS Technology

A. *Basics*

PSSS uses for the encoding m-sequences in parallel. Equation (1) describes the base m-sequence ms_1.

$$ms_1 = (m_{11}, m_{21}, ..., m_{M1}) \quad (1)$$

The coding table is given by EN and contains cyclic shifted m-sequences of ms_1.

$$EN = \begin{bmatrix} m_{11} & \dots & m_{1N} \\ m_{21} & \dots & m_{2N} \\ m_{M1} & \dots & m_{MN} \end{bmatrix} \quad (2)$$

For the encoding the data D (3) is multiplied with EN (2).

$$D^T = (d_1, d_2, ..., d_N) \quad (3)$$

$$S = EN \cdot D \quad (4)$$

Each data bit of D is spread with a cyclic shifted m-sequence. The spreaded bit are added column wise. The decoding can be reached by cyclic cross correlating the PSSS-Symbol S with the base m-sequence ms_1. This operation is similar to using a matrix DE for decoding.

$$DE = EN^T \quad (5)$$

$$CCF = S \cdot DE \quad (6)$$

CCF presents the cyclic cross correlation between the PSSS symbol S and the decoder matrix DE. The reconstruction is done by threshold decision as described in (7).

$$d'_n(ccf_n) = \begin{cases} d'_n = 0;\ ccf_n \le (Max\{CCF\} + Min\{CCF\} \div 2) \\ d'_n = 1;\ ccf_n > (Max\{CCF\} + Min\{CCF\} \div 2) \end{cases} \quad (7)$$

$d'_n(ccf_n)$ is the reconstructed data word. Depending on implementation targets of PSSS different threshold algorithms are available.

For reducing the PAPR (Peak to Average Power Ratio) and the DC component of the PSSS symbol S precoding could be used. The precoding of one symbol is executed independent of the precoding of any other symbol with the two steps described mathematically as follows:

$$S'(m) = S(m) + \frac{(Max + Min)}{2} \quad (8)$$

where S(m) is the current PSSS symbol and S'(m) is the aligned symmetric to zero PSSS symbol and Max and Min are the maximum and minimum chip amplitudes within the symbol respectively and

$$p''(m) = \frac{p'(m)}{A} \quad (9)$$

where A = (Max' – Min') and Max' and Min' are the maximum and minimum chip amplitudes within the aligned symmetric to zero PSSS symbol p'(m) respectively.

Precoding reduces the PAPR and therefore the required linearity of the power amplifier.

B. PSSS for the IEEE 802.15.4-2006 sub 1 GHz PHYs

Target for the new standard [2] was to reach 250 kbps for the sub 1 GHz PHY. For the PHY a 31 chip long sequence was selected as base m-sequence. From the resulting encoding matrix only a subset has been selected. Available are 31 cyclic shifted sequences. For FCC only five and for ETSI twenty sequences have been selected. This ensures that for the given chip rate a data rate of 250 kbps is realized, for both the FCC and ETSI versions of the PHY.

Selecting a subset of EN (4) causes the distance between the correlation peaks of CCF (6) to increase, which can be used for enhanced multipath fading robustness. The delayed multipath fading parts of the received signal are between the correlation peaks and don't cause ISI (Inter Symbol Interference), if the delay spread is shorter than the distance between the CCF peaks.

To avoid that the cyclic correlation of the decoder is hurt by multipath fading, the PSSS symbol S is cyclicly extended, similar to the cyclic extension of OFDM symbols. The extended PSSS symbol contains 32 chips.

III. PERFORMANCE OF PHY IMPLEMENTATIONS AND AVAILABLE PLATFORM

For the ETSI and FCC PHYs of IEEE802.15.4-2006 discrete FPGA based implementations are available that have a sensitivity of better than -100dBm for 1% PER. The discrete module is shown in figure 3. The available link budget is about 120 dB or even more. Nguyen et al. [3] describe a single-chip implementation in CMOS.

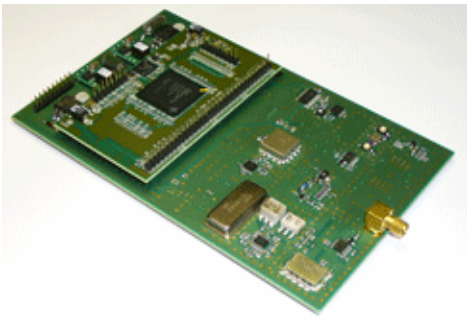


Figure 3. DWW IEEE802.15.4-2006 PSSS 868MHz "TRx154b_Eval" with Spartan 6

To evaluate the performance of the PSSS based solution a real word test was made. Figure 4 shows the urban test environment. The red dot marks the position of the transmitter in the basement of the building. The building has reinforced concrete ceilings, 50cm solid walls and the basement windows are with bars. The Tx position was 75 cm from the ground. The receiver test points are marked in yellow.

The tested modules are:

- **DWW IEEE802.15.4-2006 PSSS 868MHz "TRx154b_Eval" module**
  Transmit Power 0 dBm and +10 dBm. Even meets ETSI mask @+15dBm. 250 kbps data rate. Real PER testing.

- **868MHz Wireless M-Bus module** Transmit Power 0 dBm. Data rate 16.38 kbps. Real PER testing.

- **IEEE802.15.4-2003 868MHz module** Transmit Power +16 dBm. 20 kbps data rate. Real PER testing.

- **IEEE802.15.4-2006 2.4GHz module** Transmit Power +5 dBm. Data rate250 kbps. Only connection loss could be tested. Real testing was not implemented.

Test condition for successful coverage was PER < 1% for same packet lengths as defined in IEEE802.15.4-2006.

Figure 4 shows the tested coverage for the 868MHz Wireless M-Bus module, figure 5 the IEEE802.15.4-2003 868MHz module, figure 6 the IEEE802.15.4-2006 2.4GHz module, figure 7 the DWW IEEE802.15.4-2006 PSSS 868MHz module TRx154b_Eval with 0 dBm transmit power and figure 8 the same with 10 dBm transmit power.

Figure 4. Coverage of the 868MHz Wireless M-Bus module, dBm transmit power, 16.38 kbps. Map source Google Earth.

Figure 5. Coverage of the IEEE802.15.4-2003 868MHz module with +16 dBm transmit power and 20 kbps. Map source Google Earth.

Figure 6. Coverage of the IEEE802.15.4-2006 2.4GHz module with 5 dBm transmit power and 250 kbps. Map source Google Earth.

Figure 7. Coverage IEEE802.15.4-2006 PSSS 868MHz "TRx154b_Eval" module 0 dBm Tx power and 250 kbps data rate. Map source Google Earth.

Figure 8. Figure 8. Coverage IEEE802.15.4-2006 PSSS 868MHz "TRx154b_Eval" module 10 dBm Tx power and 250 kbps data rate. Map source Google Earth.

The conclusion of the benchmarking is, that the PSSS based module has at the same transmit power and at much higher data rate than the competitors a unique coverage. All other modules were not able to communicate across the street. A usage for metering applications seems to be difficult.

The coverage advantage of the PSSS based module can also be used to further reduce the Tx power for reducing the power consumption. The PSSS advantage is mainly caused by the enhanced PSSS robustness against multipath fading. That robustness was the selection criterion for the PHY selection at the IEEE standardization process of IEEE802.15.4-2006 .

IV. FUTURE STEPS

The PSSS solution will soon be available as a single chip based module. The chip partner is IHP GmbH in Germany. The PSSS technology is advantageous for WSN due to low power consumption and low cost of implementation, combined with unique data rates of 250 kbps for the ETSI region of the IEEE 802.15.4-2006 standard in the sub 1GHz band.

The low complexity of PSSS implementations is opening the path to high data rate solutions, where OFDM implementations are limited in the reachable data rate. PSSS can be combined with well known technologies like deconvolution and MIMO for enhancing the multipath fading robustness. Also a combination of PSSS and OFDM seems to be promising [4]. Actual R&D activities are for 100 Gbps wireless implementations of PSSS with deconvolution.

Using PSSS with deconvolution can compensate multipath fading nearly perfectly. In strong multipath fading environments decreases the PSSS performance only slightly (about 0.5dB for BER <1e-6) when using deconvolution compared to the performance in non multipath fading environment. The decrease was caused in that simulation due to the non ideal channel estimation with noise limitation.

That underlines that PSSS with deconvolution offers high performance combined with low implementation complexity. First real world tests show that the simulated performance can be realized in hardware implementations.

REFERENCES

[1] [1] A. Wolf, PSSS Patents EP04701288.5-1515/1584151, DE 10 2004 033 581, US 20060256850.

[2] H. van Leeuwen and A. Wolf, IEEE15-04-0121-03-004b, March 2004.

[3] Trung-Kien Nguyen et al., "Low-Power Direct Conversion Transceiver for 915 MHz Band IEEE 802.15.4b Standard Based on 0.18 µm CMOS Technology", ETRI Journal, Volume 30, Number 1, February 2008, pp. 33-45.

[4] Paulo Isagani M. Urriza and Joel Joseph S. Marciano Jr., "A Flexible OFDM Spread Spectrum System Using Parallel Sequences", http://eee.upd.edu.ph/urriza/research.html .

[5] H. van Leeuwen and A. Wolf, IEEE15-05-0205-05-004b, March 2005.

The development of the discrete prototypes was supported by a grant from: Europäische Union, Europäischer Fond für regionale Entwicklung. Investition in Ihre Zukunft. Allgemeine Informationen unter www.efre.brandenburg.de.

Holistic Packet Statistics for Neighborhood Management in Sensor Networks

Sebastian Ernst, Christian Renner, Christoph Weyer, and Volker Turau
Institute of Telematics
Hamburg University of Technology
Hamburg, Germany
{sebastian.ernst,christian.renner,c.weyer,turau}@tu-hamburg.de

***Abstract*—Knowledge of neighboring nodes is a fundamental requirement of many algorithms in the field of wireless sensor networks. The neighborhood relation of nodes is defined by their ability to communicate directly and not by their proximity. Due to the temporal changes of the wireless channel, link qualities must be continuously estimated. In this paper a new approach based on holistic packet statistics is presented. Unlike existing estimation techniques it forswears from squeezing a link's characteristics into a single value. The benefit of this approach is substantiated by an evaluation utilizing real-world and synthetic data.**

I. Introduction

In a variety of applications small, sensor-equipped computers (sensor nodes) collect and share data via radio in networks, which are called wireless sensor networks (WSN). Many algorithms require knowledge about the direct neighbors of each node. This information is used, e.g., for routing decisions or group communication. The information about neighbors is provided by neighborhood management protocols. The latter choose the most important and reliable nodes from the set of available nodes. One important criterion to determine the importance of a node is the link quality, which in turn is provided by a link estimator. Accuracy and agility are two important performance measures of a link estimator. The faster the link qualities are determined, the sooner the network reaches an operational state. Only with a certain amount of accuracy a reliable comparison of link qualities is possible. One approach to estimate the link quality is to use the Received Signal Strength Indication (RSSI). While this is the fastest method—only one packet is needed—,an accurate estimation is not possible using the RSSI alone [1], [2]. Another approach is to estimate the Packet Reception Rate (PRR), which represents the percentage of received packets and is sometimes referred to as Packet Success Rate (PSR). It takes less retransmissions till a packet is successfully transmitted using a link with higher PRR as compared to one with lower PRR.

In the following the results of previous research is examined, a fast and accurate estimator with a built in accuracy measure is presented and evaluated. For the evaluation of estimators a Java program is introduced.

II. Related Work

Link quality estimation has been frequently addressed by researchers. Srinivasan and Levis found in their research [2] that if the RSSI rises above a certain level, the link can be considered to be good with a PRR greater than 85%. Links with a lower RSSI however cannot be reliably identified as their PRR varies in the whole spectrum. This inaccuracy disqualifies the RSSI as an appropriate link quality measure.

Likewise the Link Quality Indicator (LQI), which also utilizes the signal strength, does not correlate with the PRR directly. The mean of the LQI does correlate with the PRR [2], but for this approach again several packets are needed. This eliminates the advantage over calculating the PRR directly.

The underlying neighbor discovery protocol of the routing protocol Topology Dissemination Based on Reverse-Path Forwarding (TBRPF) uses the PRR for a link quality measure [3]. TND (TBRPF neighbor discovery) utilizes a windowed approach to calculate the PRR, wherein the last r of s packets have to be received in order to assume that a reliable connection from sender to receiver is possible. A packet loss is detected by the serial numbers of consecutively received packets. In addition a time-out is used. This method has two major drawbacks. The first is that all packets are equally weighted, although they represent the link at different time instances in the past. The second is that a large value of s leads to slow estimations, whereas a small value of s results in a tremendous influence of each packet on the PRR, leading to a rapidly changing link quality measure.

Using an Exponentially Weighted Moving Average (EWMA) to estimate the PRR leads to a stronger weight of recent packets. The Adaptive Link Estimator (ALE) [4] uses an EWMA with adaptable weights depending on the quality of the link. To be able to estimate the link quality quickly, ALE starts with a strong emphasis on recent packets. When the PRR reaches a certain level the weights are shifted in order to get a more accurate estimation. If the PRR drops below a certain level the weights are shifted back again. To avoid oscillation of the weights, a hysteresis is defined. This leads to fast and (mostly) accurate estimations, but it also creates new problems. The oscillation of the PRR due to the starting weights demands for a rather large hysteresis, such that the weights will not be shifted to a slower estimation by accident. On the other hand, if a link changes from good to bad (e.g., due to a change of weather or newly appeared obstacles), it will take more time to recognize this change with a large hysteresis.

III. Holistic Packet Statistics

Previous research shows that link-quality estimation is an important, non-trivial topic. It is known that wireless link qualities are prone to change over time [1], [5]. Previous research concentrated on creating one fast and accurate estimator using a single value that represents the percentage of sent packets that actually succeed.

However, the main drawback of this approach is the absence of knowledge about link dynamics in the recent past, i.e., there is no information on the course of the PRR. Consider a link with a PRR of 0.9. There are basically three different possibilities of its past: The link could be stable, it may have failed but is currently recovering; or it was very good, but is now decreasing. This example reveals the need for a more fine-grained link-quality metric.

For this reason, the Holistic Packet Statistics (HoPS) distinguish between short- and long-term link quality plus a metric for describing the fluctuation of the latter. While the short-term link quality depicts the current state of the link and is therefore able to detect short disturbances, the long-term link quality represents the state of the link for a longer period of time. The fluctuation is split into the reliability of the link as well as an indicator for changes in long-term link quality. In the following the link quality measures used by HoPS are defined. Afterwards a way to speed up the start of the estimation is presented.

A. Link Quality Measures

1) Short-term Estimation: The PRR is recursively estimated and stored as short-term link quality

$$\xi_t = \alpha\xi_{t-1} + (1-\alpha)P_t \;\;, P = \begin{cases} 1 & , \text{ packet received} \\ 0 & , \text{ packet missed} \end{cases} \quad (1)$$

Resolving the recursion shows that this leads to α being the exponential weighting factor, which is to be chosen such that $0 \leq \alpha \leq 1$. A smaller α leads to a stronger emphasis of recent packets:

$$\xi_t = (1-\alpha)\sum_{i=0}^{n} \alpha^i P_{t-i} + \alpha^{n+1}\xi_{t-(n+1)} \quad (2)$$

2) Long-term Estimation: The long term-link quality is a second order smoothing of the PRR:

$$\nu_t = \beta\nu_{t-1} + (1-\beta)\xi_t \quad (3)$$

The special case of $\alpha = \beta$ is used in economics to predict the next iteration of share prices, but this would either slow down ξ or speed up ν. The influences on the predictability with $\alpha \neq \beta$ go beyond the scope of this paper.

3) Fluctuation Estimation: In order to determine the amount of oscillation and the trend of ν, the linear lower and upper deviation are defined, using the estimated mean $\tilde{\mu} = \mu + \Delta\mu$ where $\Delta\mu$ is the estimation error and μ the actual arithmetic mean value.

$$\delta^- := \frac{1}{n}\sum_{x^- \in X^-} (\tilde{\mu} - x^-) \;, \;\; \delta^+ := \frac{1}{n}\sum_{x^+ \in X^+} (x^+ - \tilde{\mu}) \quad (4)$$

with X being a set of values and

$$\mu = E\{X\}, X^+ = \{x \in X | x > \tilde{\mu}\}, X^- = X \backslash X^+, n = |X|$$

It can be shown that the estimation error is the difference of these deviations:

$$\begin{aligned} \delta^+ - \delta^- &\overset{(4)}{=} \frac{1}{n}\sum_{x\in X}(\tilde{\mu} - x) = \frac{1}{n}\sum_{x\in X}(\mu + \Delta\mu - x) \quad (5) \\ &= \underbrace{\frac{1}{n}\sum_{x\in X}(\mu - x)}_{=0} + \frac{1}{n}\sum_{x\in X}\Delta\mu = \Delta\mu \end{aligned}$$

This shows how the linear lower and upper deviation can be used to determine the error being made while estimating the mean of a set.

Similarly, the sum of δ^+ and δ^- results in the average absolute deviation.

$$\begin{aligned} \delta^+ + \delta^- &= \frac{1}{n}\left(\sum_{x\in X^+}(x - \tilde{\mu}) + \sum_{x\in X^-}(\tilde{\mu} - x)\right) \quad (6) \\ &= \frac{1}{n}\sum_{x\in X}|x - \mu| \quad (7) \end{aligned}$$

Notice that the estimation error does not have an effect on the absolute deviation.

To determine the average absolute deviation of ξ and the trend of ν, the linear lower and upper deviations are recursively estimated:

$$\begin{aligned} \delta_t^+ &= \gamma\delta_{t-1}^+ + (1-\gamma)\varphi(\xi_t, \nu_t) \quad (8) \\ \delta_t^- &= \gamma\delta_{t-1}^- + (1-\gamma)\varphi(\nu_t, \xi_t) \quad (9) \\ \varphi(a,b) &= \begin{cases} a-b & , \text{ if } a > b \\ 0 & , \text{ else} \end{cases} \quad (10) \end{aligned}$$

The recursive estimation of the deviations introduce another estimation error, so that the difference $\delta^+ - \delta^-$ yields a measure for, rather than the exact estimation error $\Delta\mu$. The same holds true for the sum $\delta^+ + \delta^-$ and the amount of oscillation accordingly. An exact calculation of the deviations would imply a huge memory demand. As resources of sensor nodes are usually very limited, an exact calculation is not practical.

B. Reduction of Start-up Time

One of the most important phases in link quality estimation is the beginning. In recursive estimation a seed is needed to start the estimation. This leads to potentially large errors in the very beginning. To further speed up the estimation in this phase, a blend is introduced for the weights of ξ, such that the weights emphasize the near past instead of the past introduced by the seed.

In the beginning ξ does not represent the actual link quality. Hence, this phase is excluded from the estimation of ν. This is achieved by setting $\nu = \xi$ in this phase. As the stability of the link and therefore the stability of ξ is not known in the beginning, the weights for ν are also blended. Otherwise a comparison of two different connections via their ν would not be possible due to their different initial conditions.

IV. Evaluation Software

To visualize the results of HoPS a Java program with GUI was implemented. A screen shot of the GUI is shown in Fig. 1. The program can be used to compare the results of different estimators. It shows five plots to display the link quality measures of the estimators. In addition a histogram shows the consecutively lost packets for analysis of the link. The real link quality can be modeled with a windowing function. On the left-hand side the parameters of the estimators can be modified. Different error measures are implemented to compare the deviation of the estimators to the reality model chosen. The plotted x-range can be adjusted to focus on certain parts of the connection. The calculated data can be exported to a file, e.g., to be used with Gnuplot.

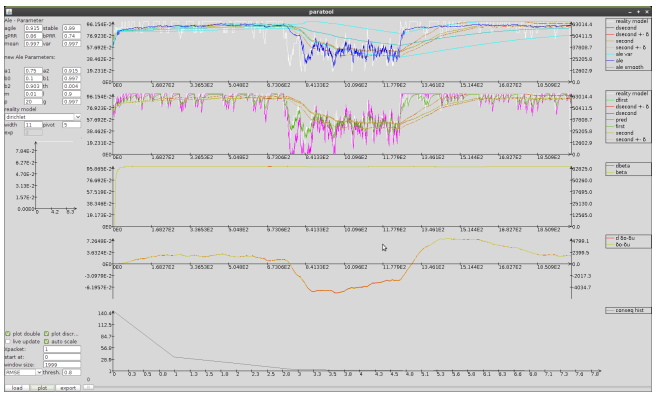


Fig. 1. Evaluation Software

A. Implementation of HoPS

So far all formulae presented are based on floating point arithmetic. Yet, the hardware of some platforms cannot handle floating point numbers. Floating point arithmetic on these platforms is performed in software, needing a lot of resources. Therefore, we implemented a version of HoPS based on 16 bit unsigned integers. Since values in the range of 0 to 1 are more illustrative than in the range of 0 to 65535, most results shown are from a floating point implementation as the implementation using unsigned integers does not differ significantly from the implementation using floating point arithmetic, which is only shown briefly.

V. Evaluation

A. Testbed

At the Institute of Telematics at the Hamburg University of Technology IRIS nodes from Crossbow running TinyOS 2.1 are used. Real data has been collected using seven nodes placed in four rooms in an office environment illustrated in Fig. 2. The closest nodes were less than 2 m apart and the largest distance between two nodes was approx. 40 m. Most nodes were separated by several walls and other obstacles, resulting in real life permanent and non permanent disturbances. Each node was connected by wire with a computer which served as data sink. For 13 days each node broadcasted one

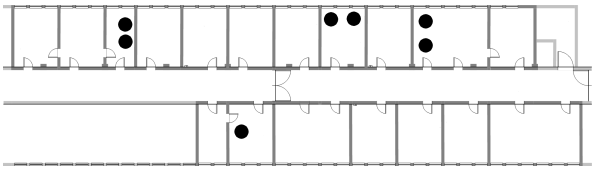


Fig. 2. Position of nodes (black dots) in the Telematics office environment

packet every second. Each node randomly chose a point in time in each second for its broadcast. A packet had a payload size of 10 Byte resulting in a packet size of 23 Bytes. The payload contained only a sequence number and two unused fields for the receiver and the sender. On receiving a packet, a node filled the fields for receiver and sender and sent the packet to the computer, where it was saved in a log file.

These log files and artificially generated log files are the basis for the evaluation of HoPS.

B. Parameters

To evaluate HoPS the results are compared with an existing estimator. ALE, trying to give an agile and accurate estimator using similar techniques as HoPS, is the most suitable one.

As proposed in [4] the agile and stable weights of ALE are 0.915 and 0.99 respectively. The hysteresis for the weights is 0.06 around 0.8 for the same reason. The estimated PRR has to rise above 0.86 or drop below 0.74 to shift the weights for ALE.

For smooth transitions the blends for the weights of ξ and ν are implemented as follows:

$$\alpha = \alpha_1 - \left\lfloor \frac{a}{age+1} \right\rfloor \frac{(\alpha_1 - \alpha_0)}{a} \quad \begin{matrix} \alpha_0: & \text{initial } \alpha \\ \alpha_1: & \text{final } \alpha \\ a: & \text{blend size} \\ age: & \text{nodes age} \end{matrix} \tag{11}$$

$$\beta = \begin{cases} 0 & , age < \frac{a}{2} \\ \beta_1 - \left\lfloor \frac{2a}{age+1} \right\rfloor \frac{(\beta_1 - \beta_0)}{2a} & , age \geq \frac{a}{2} \end{cases} \tag{12}$$

The estimation of the short-term link quality starts with a weight of $\alpha_0 = 0.75$ which reaches $\alpha_1 = 0.915$ after $a = 20$ packets as defined previously. For the estimation of the long-term link quality the weights start at $\beta_0 = 0.1$ and increase thereafter to $\beta_1 = 0.997$. Note that due to the delay β starts practically at 0 and has a total blend time of $2a$. The deviations are estimated with $\gamma = 0.997$.

C. Analysis

To illustrate the oscillation of ALE's weights, a part of a real log file is shown in Fig. 3. Due to the oscillation of ALE when using the agile weight, the threshold of 0.86 is crossed, which leads to an overestimation of the link's future quality.

A synthetic link with disturbance is shown in Fig. 4. Each packet was received with a probability of 95%, except for the packets with numbers 750 to 800, which were lost entirely. A center pivoted rectangular window of width 10 was used to model the real link quality. It can be seen that ξ drops very quickly, indicating the lossy link. On the other hand ν,

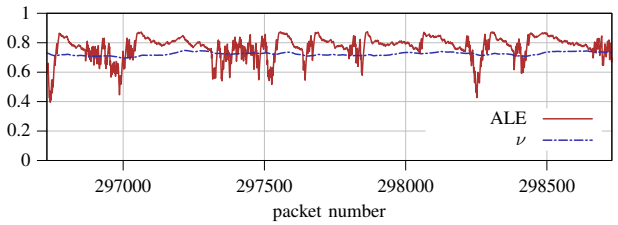
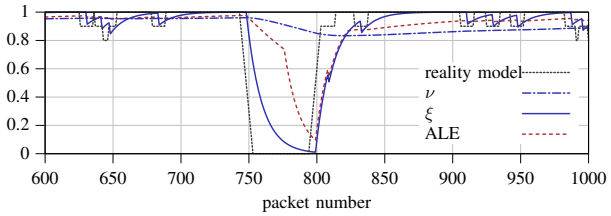
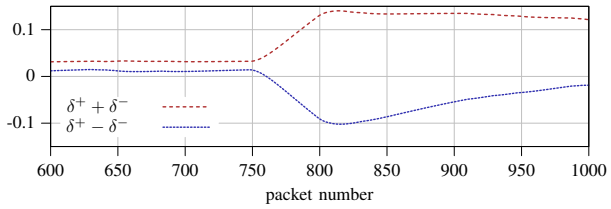


Fig. 3. Comparison of ALE and the long term link quality estimation with 2000 packets of a real link where 72.5% arrived at their destination



(a) Estimated PRR



(b) Upper and lower deviation

Fig. 4. Synthetic link with a disturbance of 50 consecutively lost packets

representing the long-term link quality, is not affected at first and than drops with increasing slope. Using only ξ and ν a change of link quality can be determined here. By including the deviations in the observation it can be determined how grievous the disturbance is at each point in the course of the disturbance.

After the disturbance ξ is greater than ν, but $\delta^+ - \delta^-$ is still negative due to the recursive estimation. Combining these three findings indicates the previous disturbance. In theory $\delta^+ - \delta^- < 0$ means a decrease of link quality, but in this situation the contrary is the case. To determine the state of the link, all four quality measures are needed. Close to the crossing of ξ and ν the situation cannot be determined uniquely. Also notice that $\delta^+ - \delta^-$ decreases faster than $\delta^+ + \delta^-$, which indicates lower estimation error with larger average absolute deviation.

Figure 5 shows that ν using floating point arithmetic (left y-axis) and $\tilde{\nu}$ using 16 bit integer arithmetic (right y-axis) behave essentially the same. Detailed tests showed that for very stable links, i.e., for $\delta^+ - \delta^- < 10^{-3}$, there are minor differences in magnitude of the deviations δ^+ and δ^-. As these effects do not compromise the comparability of two different links and do not change the meaningfulness of the deviations, these effects are negligible.

Another effect depicted in Fig. 5 is the influence of blends

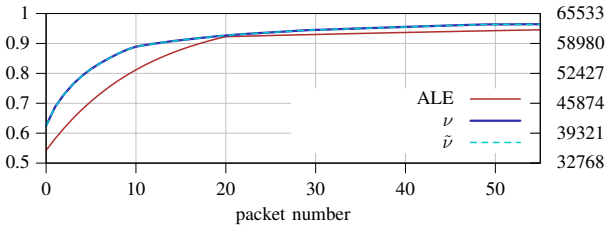


Fig. 5. Comparing rise time of ALE and HoPS

on the rise time. As compared to ALE ν rises faster. While further decreasing α_1 leads to even faster responses, it increases the likelihood of overshoots, leading to inaccuracy.

The figures shown here, especially Fig. 3 and 4(a), show that a fast and accurate estimation of link quality using one quality measure only is not possible. Using different link quality measures enables HoPS to achieve just that. With a careful analysis of the four measures presented it is possible to make a statement about the holistic condition of a link.

The higher computing expenditure and the need for more memory are the drawbacks. Having three extra link quality measures means 6 Byte extra memory for each link, if 16 Bit unsigned integer values are used. However, the extra information gained outweights the extra memory.

VI. Conclusion

In this paper a new approach to link quality estimation based on holistic packet statistics is developed and assessed. This new technique reveals itself to be an agile yet reliable estimator with link quality measures that surpass plain PRR estimations. The latter statement is supported by an evaluation using real-world and synthetic data. The increased expressiveness of a holistic view onto link quality enables a more detailed and accurate judgment. In consequence, application-specific considerations for neighborhood relationships can be implanted more easily. This advantage comes at the cost of an increased memory footprint plus extra calculations. However, the benefits of a holistic statement about the link quality predominates, plus these costs are relatively small compared to the resources of modern sensor nodes.

References

[1] A. Woo, T. Tong, and D. Culler, "Taming the Underlying Challenges of Reliable Multihop Routing in Sensor Networks," in *Proceedings of the First International Conference on Embedded Networked Sensor Systems (SenSys'03)*, Los Angeles, CA, USA, Nov. 2003.

[2] K. Srinivasan and P. Levis, "RSSI is Under Appreciated," in *Proceedings of the Third Workshop on Embedded Networked Sensors (EmNets'06)*, Cambridge, MA, USA, May 2006.

[3] R. Ogier, F. Templin, and M. Lewis, "Topology Dissemination Based on Reverse-Path forwarding (TBRPF)," 2004.

[4] C. Weyer, S. Unterschütz, and V. Turau, "Connectivity-aware Neighborhood Management Protocol in Wireless Sensor Networks," in *Proceedings of the 7th GI/ITG KuVS Fachgespräch "Drahtlose Sensornetze" (FGSN'08)*, Berlin, Germany, Sep. 2008.

[5] V. Turau, M. Witt, and C. Weyer, "Analysis of a Real Multi-hop Sensor Network Deployment: The Heathland Experiment," in *Proceedings of the Third International Conference on Networked Sensing Systems (INSS'06)*, Chicago, IL, USA, Jun. 2006.

Desynchronization in Multi-Hop Topologies: A Challenge

Clemens Mühlberger
Chair of Computer Science V
Department of Computer Science
University of Würzburg, Am Hubland, 97074 Würzburg
Email: muehlberger@informatik.uni-wuerzburg.de

***Abstract*—The biologically inspired primitive *desynchronization* was successfully implemented and tested within single-hop topologies in form of the self-organizing TDMA protocol DESYNC for wireless sensor networks (WSNs). Two extensions of that MAC protocol for multi-hop topologies have been discussed, but either the extended protocol is not all-purpose but specified for just a specific subset of multi-hop topologies, or each node has to broadcast all of its neighboring information at every single packet, which enlarges the packet size and thus consumes additional energy and bandwidth. One reason for this limitation, and packet overhead respectively, is the hidden terminal problem which is inherent in all multi-hop topologies. In this paper we compare the characteristics of single-hop and multi-hop topologies – with respect to the primitive of desynchronization. We will further analyze one special multi-hop topology in detail, which not only shows the complexity of a multi-hop desynchronization, but also provides new opportunities to support all sorts of multi-hop topologies with reduced overhead for the neighboring information.**

I. DESYNC – A BRIEF INTRODUCTION

In 2006, Degesys et al. [1] first published DESYNC, a self-organized TDMA protocol for WSNs [2]. This MAC protocol follows the biologically inspired primitive of *desynchronization* [3] to achieve an equidistant distribution of participating oscillators, e.g. periodically transmitting sensor nodes. As logical opposite of synchronization, desynchronization in general means that each device tries to perform its (periodic) tasks as far away as possible from all other affected devices. Within the scope of WSN, desynchronization describes the temporally equidistant transmission of radio packets.

So, the (idealized) network is composed of a set of nodes N. All communication links are symmetric, and each node $i \in N$ oscillates at an identical frequency. The phase ϕ_i of a node i denotes the elapsed time since its last transmission relative to its current period. When a node finishes its period, it broadcasts a so called *firing packet* and immediately resets its phase, i.e. if the node is desynchronized already, it will broadcast its next firing packet exactly one period after the start of the current transmission. Each one-hop neighbor of the currently transmitting node receives this firing packet (if there was no collision), and logs the sender's ID together with its local time of reception to calculate its individual phase shift towards the sender.

Each node i can determine by itself a more appropriate firing phase (according to an equidistant distribution), based on its individual knowledge of the phases of its so called *phase neighbors*:

- *previous phase neighbor* (predecessor) $p(i) \in N \setminus \{i\}$ broadcasts its firing packet (from i's point of view) just before node i,
- *successive phase neighbor* (successor) $s(i) \in N \setminus \{i\}$ broadcasts its firing packet (from i's point of view) just after node i.[1]

With it, node i can now calculate the midpoint of its phase neighbors, and finally estimate its new firing phase ϕ_i' as

$$\phi_i' = (1-\alpha) \cdot \phi_i + \alpha \cdot \frac{\phi_{s(i)} + \phi_{p(i)}}{2}, \qquad (1)$$

where ϕ_i denotes the last phase of node i, and the *jump size parameter* $\alpha \in (0.0, 1.0]$ regulates, how fast the node moves toward the assumed midpoint of its phase neighbors. Convergence to the stable state of *desynchrony* is achieved, if each node has the same distance to its phase neighbors (cf. Fig. 1) and thus the transmission times do not change anymore - unless the system changes.

II. EXTENSIONS FOR MULTI-HOP TOPOLOGIES

The handling of single-hop topologies is quite simple[2], because every node can directly communicate with each other. On the other hand, the so called *hidden terminal problem* inheres in multi-hop topologies, which complicates collision-free communication. This section presents two yet available extensions of the DESYNC protocol for multi-hop topologies: M-DESYNC and EXTENDED-DESYNC.

A. The M-DESYNC Approach

The M-DESYNC algorithm [4] for (single-hop and) acyclic multi-hop topologies is mainly based on the *local max degree* of each node, i.e. the maximum degree among a node i and its one-hop neighbors $N_1(i)$. Here, the *degree* of a node equals the cardinality of its one-hop neighborhood $|N_1(i)|$.

This algorithm requires an initial phase, at which each node exchanges its degree with all its one-hop neighbors to determine its local max degree. This phase may take quite long, because the algorithm uses just a random back-off protocol

[1] Besides, within a connected topology of size $|N| = 2$ both phase neighbors are the very same node $p(i) = s(i)$.

[2] This statement will be confirmed in detail in Section III-A.

without further optimization. After this preliminary phase, every node requires local max degree plus an additional time slots for a collision-free communication within its interference range. At the next step, each node just has to occupy its individual time slot. For this slot selection, a modulo precoloring as well as a priority-based strategy are suggested instead of just a random competition.

Using the local max degree, the minimum number of required time slots per period for each node was proven. However, the M-DESYNC approach is not very flexible to topology changes due to the lengthy exchange phase, but even not applicable for cyclic topologies, which will be demonstrated in Section IV.

B. The EXTENDED-DESYNC Approach

To solve the hidden terminal problem at multi-hop topologies, each node needs knowledge about its two-hop neighborhood. Therefore, for the EXTENDED-DESYNC algorithm [5] each node broadcasts its (currently known) one-hop neighbors in combination with their relative phase shifts, always corresponding to the point of view of the current sender. With it, each node gets to know its two-hop neighborhood in addition. The relative phase shifts may become stale, because phase changes of two-hop neighbors emerge after two periods. But this delayed information becomes more accurate and reliable with each subsequent period and thus just slows down convergence rate a little.

Here, no initial exchange phase is required. Instead, a new joining node just has to listen for a few periods to make itself familiar with its local topological conditions. Afterwards, it can interact immediately with its well-known one-hop neighbors and thus be integrated into the network easily.

Hence, the EXTENDED-DESYNC approach is very flexible and reacts quite fast on topology changes. It thus scales well with network size, but exhibits a large packet overhead. Every node has to broadcast its whole one-hope neighborhood, which takes bandwidth and energy for algorithmic purposes, especially in dense networks and at nodes with a high degree.

III. COMPARISON

Before we oppose the characteristics of desynchronization in single-hop topologies to multi-hop topologies, we specify some general assumptions. For all nodes we assume that their communication range equals their interference range. Next, the network is build upon symmetrical links, i.e. communication between two nodes always works bidirectional. And finally, the network consists of $|N|$ nodes, where every node owns a unique identifier as well as a finite buffer for storing (incoming) packets. But each node has just one transceiver in half-duplex mode, i.e. no node can transmit and receive packets simultaneously.

A. Single-Hop Topology

Within a single-hop topology, every node is able to interact with each other, hidden nodes do not exist. Thus, everyone knows everyone, each node has knowledge about the whole network. This enables a fast and easy self-adaption on start-up and topological modifications. Furthermore, all nodes share the common communication medium, that means for a desynchronized TDMA protocol there are exactly $|N|$ slots required at every period.

In single-hop topologies, a packet transmission is considered to be successful, if there are no other packet transmissions at the same time, i.e. the shared communication medium is assumed to be error-free. Thus, at every point in time just one single node is allowed to send a radio packet. In terms of desynchronization, the stable state (*desynchrony*) is reached, if each node transmits its radio packets temporally equidistant to its phase neighbors. With it, we can draw the following conclusions for desynchronization in single-hop topologies.

- S1 All nodes within a single-hop topology have the very same degree $|N| - 1$.
- S2 If node i is phase neighbor (w.l.o.g. predecessor $p(k) = i$) of another node $k \in N \setminus \{i\}$, then node k in return is the corresponding phase neighbor (here, k is successor $s(i) = k$) of i.
- S3 Every node i with degree ≥ 1 has at most one predecessor $p(i) = j$ and at most one successor $s(i) = k$. Following from S2, node i will be the corresponding phase neighbor (successor, and predecessor respectively) of its phase neighbors j and k in return.
- S4 Using S3, every node i with degree ≥ 1 is always predecessor $p(j) = i$ and successor $s(k) = i$ of nodes $j, k \in N \setminus \{i\}$.
- S5 Due to S2 and according to equation 1 (every node tries to maximize the temporal distance to both its phase neighbors), all nodes are distributed equidistant along the unified period. In other words, the temporal distances between each pair of subsequently firing nodes are identical.
- S6 The initial start-up order determines, when a node will (re)join or leave the network, mainly affects the order of firings.

B. Multi-Hop Topology

Within a multi-hop but connected topology, there exists at least one node i which is not able to interact with every node $j \in N \setminus \{i\}$ of the network in a direct way. For this reason, there exists at least one such ”hidden” node $h \in N \setminus \{i\}$ outside the communication range of node i. Hence, every node has just a local view and thus limited knowledge about the whole network. Although all nodes share the same communication medium. Indeed, it will be possible now, that two or more nodes can transmit their packets simultaneously within the same time slot without interference. Therefrom, for a desynchronized TDMA protocol at most $|N|$ transmission slots are required to support a collision-free communication within the network.

This is the reason, why a packet transmission is considered to be successful, if there are no other packet transmissions at the same time within the interference area of the sender and all of its potential receivers. Thus, more than one node may be

allowed to transmit a radio packet concurrently. Desynchrony is reached here, if each node transmits its packets temporally equidistant to its phase neighbors without interference with any other node of the network. For desynchronization in multi-hop topologies the following phenomena can be observed:

- M1 The degree of the nodes within a multi-hop topology now may diverge, but is at most $|N| - 1$.
- M2 Due to the nodes' different degrees in multi-hop topologies (cf. M1), $s(i) = j \Leftrightarrow i = p(j)$ as well as $p(i) = k \Leftrightarrow i = s(k)$ (cf. S2) do not hold any longer for a node i and its phase neighbors $j, k \in N \setminus \{i\}$. For example, node i is predecessor $p(j) = i$ of node j, but in turn node j is not i's successor $s(i) \neq j$, but instead node $k \neq j$ is now successor $s(i) = k$ of i.
- M3 As for single-hop topologies (cf. S3), every node i with degree ≥ 1 has at most one predecessor and at most one successor. But now, in multi-hop topologies there can be a set of nodes $S = \{x|s(x) = i\} \subseteq N \setminus \{i\}$ with $|S| \geq 2$ sharing the same successor i. Analogously, there can be a set of nodes $P = \{x|p(x) = i\} \subseteq N \setminus \{i\}$ with $|P| \geq 2$ sharing the same predecessor i. Changing the firing time of such a multiple successor (and predecessor respectively) will affect at once the time of firing of every node $x \in S$, and $x \in P$ respectively, which initiates the recalculation of x's next firings and thus slows down convergence rate.
- M4 In single-hop topologies (cf. S4), every node i with degree ≥ 1 is always predecessor and successor at once. But due to observation M2, multi-hop topologies can contain nodes i with degree ≥ 1, which are either just predecessors, or just successors, or none of another node. That means, changing the time of firing (within a specific interval) of such a node i does not initiate recalculation of many (if any) time of firings, but maybe contradicts the primitive of desynchronization (cf. Sec. I).
- M5 The observation M2 of not-being phase neighbor of node's phase neighbors, linked to the availability of different degrees in multi-hop topologies (cf. M1), leads to non-identical temporal distances. That is, each node tries to maximize its temporal distance towards its phase neighbors (cf. S5), but within multi-hop topologies the temporal distance between each pair of subsequently firing nodes are not identical anymore.
- M6 As for single-hop topologies (cf. S6), the initial start-up order not only mainly affects the order of firings, but also whether a node becomes phase neighbor of other nodes – or not.

The nodes' temporal order and the phase neighbors of a node within a multi-hop topology strongly depend on the initial start-up order. Because of this large configuration space and observations M1 – M6, the proof of convergence for any kind of multi-hop topology is quite difficult – especially from an arbitrary initial start-up order into the stable state of desynchrony. To get a first impression of the difficulty of such a proof see [6]. Such a proof will be object for our future research. Therefore, we will exemplify in the next section, why cyclic topologies are not covered by the M-DESYNC approach and which information could be sufficient to get such a multi-hop topology desynchronized – always depending on the initial start-up order.

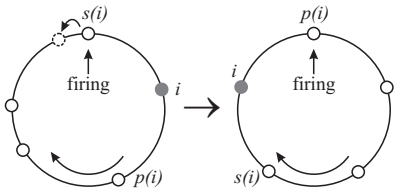


Fig. 1. Snapshots of the process of desynchronization from a global point of view.

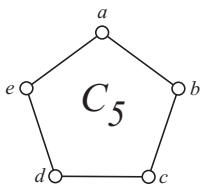


Fig. 2. Diagram of the examined topology C_5.

IV. MULTI-HOP EXAMPLE

In this section we analyze the desynchronization of a 2-regular Hamiltonian cycle C_5 of size $|N| = 5$, i.e. there are five nodes $a, \ldots, e \in C_5$, all have degree two, according to Fig. 2. For a collision-free communication within C_5, there are five time slots required: If for example node a transmits a packet, neither its one-hop neighbors e and b, nor its two-hop neighbors d and c are allowed to transmit any packet at the same time. Due to the symmetry properties, this holds for all other nodes of topology C_5. Thus, each node claims one of totally five slots.

Using the local max degree method of the M-DESYNC approach does not lead to a correct and collision-free time slot assignment by the following reasons. First, the degree of each node is two, just as any local max degree. With an additional slot for itself, each node schedules three slots in total. But for a collision-free communication within topology C_5, at least five instead of just three disjoint slots are required (see above). For this reason, the M-DESYNC algorithm is non-applicable for cyclic multi-hop topologies.

In contrast, the EXTENDED-DESYNC algorithm schedules five time slots according to the five nodes. Because each node transmits its currently known one-hop neighborhood, each node also gets to know its two-hop neighborhood. With this knowledge, each node can take care of its one-hop – and more important – of its two-hop neighbors. Due to the symmetry properties of this topology C_5, still every node has two one-hop and also two two-hop neighbors and thus schedules five slots in total. Therefore, each node desynchronizes itself according to its phase neighbors, which in turn depend on the initial network configuration (cf. M6).

To reduce the packet overhead which has to be propagated at the EXTENDED-DESYNC algorithm, we will go step-by-step through one (of many) possible desynchronization procedures for the formation of this multi-hop topology C_5, using the following packet format $[i_{id}, p(i)_{id}, s(i)_{id}, |N_1(i)|]$ which contains the following data[3]:

[3]The relative phase shift of the phase neighbors are also transmitted but not shown here to cut short the example.

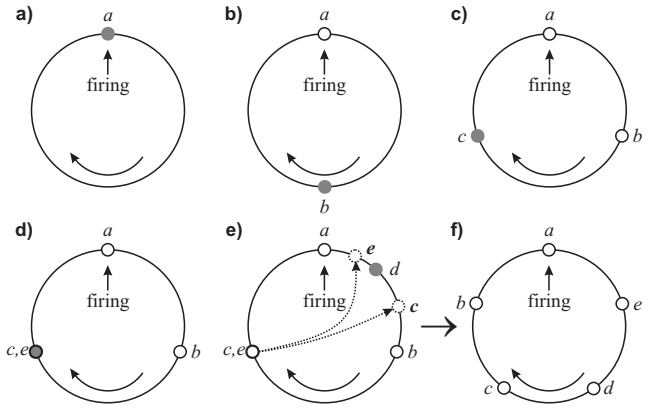


Fig. 3. Step-by-step desynchronization for topology C_5 with less overhead from a global point of view.

- i_{id}: the ID of the sender named i,
- $p(i)_{id}$: the ID[4] of the current predecessor $p(i)$ of the transmitting node i,
- $s(i)_{id}$: the ID[4]of the current successor $s(i)$ of the transmitting node i,
- $|N_1(i)|$: the current number of one-hop neighbors of the transmitting node i.

The desynchronization process for the start-up order $a \rightarrow b \rightarrow c \rightarrow e \rightarrow d$ runs as follows (cf. Fig. 3).

a) First, node a starts up and listens, but receives not a single packet. Thus after a while, node a builds a new network and broadcasts $[a,,,0]$ after every period.

b) Next, node b starts up, listens and receives $[a,,,0]$ from node a. Thus, node b broadcasts $[b,a,a,1]$. Node a in return receives b's packet and from now on broadcasts $[a,b,b,1]$ accordingly.

c) Node c wants to join the network and listens, but just receives b's broadcast. With it, node c in return broadcasts $[c,\underline{a},b,1]$. This causes node b and – with some delay – node a to adjust their time of firings, as well as the content of their packets to $[b,c,a,2]$, and $[a,b,\underline{c},1]$ respectively.

d) Later, node e starts up and listens. It receives just the broadcast of node a. Using this information, node e chooses the same time of firing as node c and broadcasts $[e,a,\underline{b},1]$. This is possible, because node e and c are currently more than two hops away and thus do not interfere with each other. The broadcast of node e causes a just to update its packet content into $[a,b,e,2]$.

e) Last but not least, node d tries to join the network and listens. Because both its one-hop neighbors e and c transmit their firing packets at the very same time, d receives just corrupt data – if any, and thus just broadcasts $[d,,,0]$. Assuming that this broadcast does not collide with any other packet, i.e. the time of firing of node d does not overlap with any time slot of the remaining network, and in this example temporally lies in between node a and b (cf. Fig. 3.d), its one-hop neighbors e and c receive d's broadcast. But because node d states not to know any neighbors (especially not e and c), each receiver concludes to cause a collision. With it, node e, and c respectively, changes its time of firing in such a way, to be in between the joining node d and its one-hop neighbor a, and b respectively. The firing packets of e and c also change to $[e,d,a,2]$, and $[c,b,d,2]$ respectively. These changes cause the corresponding neighbors to adjust their time of firing and content of their firing packets into $[a,e,b,2]$, $[b,a,c,2]$, and $[d,c,e,2]$.

f) Finally, after the nodes rearranged themselves along the period, each node holds the same distance to its both phase (and one-hop) neighbors. Remarkably, all nodes are temporally equidistant distributed, although this is not a single-hop topology.

[4]If the identifier is underlined, the corresponding node is a two-hop neighbor of sender i.

V. Conclusion and Outlook

In this paper we initially introduced the primitive of desynchronization as TDMA protocol for WSNs. We then analyzed the difference between desynchronization in single-hop and multi-hop topologies. The detailed example of desynchronization in a specific cyclic multi-hop topology on the one hand presented a collision-free slot assignment using reduced firing data. But on the other hand, this idealized example leaves many question open, e.g. what, if the transmission of node d always interferes with a's broadcast, thus d never will be received? Will the system converge for any other start-up order, too? Is the reduced firing data sufficient or too much limiting for other multi-hop topologies?

These questions are subject to our future research. Also, we plan to strengthen and to generalize the approach of reduced firing data. This may help us to prove the convergence of our reduced data approach for arbitrary multi-hop topologies in any start-up order. To get our approach fit for practice, we plan to implement it within a real-world testbed: for instance, realistic scenarios do not have symmetrical links in any case.

References

[1] J. Degesys, I. Rose, A. Patel, and R. Nagpal, "DESYNC: Self-Organizing Desynchronization and TDMA on Wireless Sensor Networks," Havard University, Cambridge, MA, USA, Tech. Rep. TR-18-06, Dec. 2006.

[2] A. Patel, J. Degesys, and R. Nagpal, "Desynchronization: The Theory of Self-Organizing Algorithms for Round-Robin Scheduling," in *SASO*, Cambridge, MA, USA, 2007, pp. 87–96.

[3] R. E. Mirollo and S. H. Strogatz, "Synchronization of Pulse-Coupled Biological Oscillators," *SIAM Journal on Applied Mathematics*, vol. 50, no. 6, pp. 1645–1662, 1990.

[4] H. Kang and J. L. Wong, "A Localized Multi-Hop Desynchronization Algorithm for Wireless Sensor Networks," in *INFOCOM*, 2009, pp. 2906–2910.

[5] C. Mühlberger and R. Kolla, "Extended Desynchronization for Multi-Hop Topologies," Institut für Informatik, Universität Würzburg, Tech. Rep. 460, Jul. 2009.

[6] J. Degesys and R. Nagpal, "Towards Desynchronization of Multi-hop Topologies," in *SASO*, Venice, Italy, 2008, pp. 129–138.

Describing Packet Payload Structures using Lightweight Semantic Data Type Annotations

(Extended Abstract)

Andreas Reinhardt, Diego Costantini, Ralf Steinmetz
Multimedia Communications Lab, Technische Universität Darmstadt
Rundeturmstr. 10, 64283 Darmstadt, Germany
Email: {andreas.reinhardt, diego.costantini, ralf.steinmetz}@kom.tu-darmstadt.de

***Abstract*—In the majority of wireless sensor networks, packet structures are statically defined at design time. At runtime, sensed data is then inserted into the payload fields prior to packet transmission. While this is efficient in terms of the required processing, the packet structure cannot be modified during runtime. However, in certain situations the need for adaptation of the packets to new requirements arises, e.g. when the energy source approaches depletion and energy-hungry sensors are deactivated to extend the node lifetime. The countermeasure of defining a multitude of packet structures to encounter any possible situation is infeasible both in terms of efforts and resource consumption.**

To address this limitation, we propose the annotation of data fields in outgoing packets by identifiers indicating the contained data types, so that any node can send payloads with dynamically defined contents. The size increase incurred by the use of annotations for each payload field can however become significant as the annotations must be sufficiently expressive to uniquely describe the payload field. To keep this size increment small, we present a supplementary approach that assigns binary aliases for the used data type annotations, thus increasing the payload space available for application data. This is especially useful as payload sizes in sensor networks are generally limited by the radio protocol, and fragmentation is expensive in terms of the according energy requirement.

I. Introduction

A common characteristic in Wireless Sensor Network (*WSN*) deployments, e.g. in environmental monitoring settings like PermaSense [1] or GlacsWeb [2], is that the structures of radio packets used in these deployments have been designed in a static manner at design time. While such static packet structures eliminate the overhead of assembling packet contents dynamically before transmission, they also hamper the adaptation to the characteristics of the sensor devices. If nodes are fitted with multimodal sensing capabilities, the transmission interval is generally determined by the sensor with the smallest sampling interval. A second observation is that the use of convergecast routing algorithms, such as the Collection Tree Protocol [3], is prevalent. Packets with sensor data are forwarded along the branches of the routing tree to its root (the *sink* node), possibly relayed by a number of intermediate nodes. Although sensor data can be aggregated while being forwarded to the sink node ([4], [5]), statically defined packets impair the applicability of in-network data processing; data aggregation on intermediate nodes can only be performed to a limited extent when the representation of the results is limited to a set of pre-defined message structures.

To overcome these limitations, we propose to extend WSN applications to support the dynamic composition of packet payloads. Obviously, the definition of the packet structure must however be present at the receiver side to correctly interpret the contents of the packet. It can either be decoupled from the packet itself and transmitted in advance (such as done in ASN.1 [6]), or alternatively be provided inline with the packet payloads. As frequent changes to the packet structures (in some scenarios, each transmitted packet might be composed differently) would incur a great number of updates, we have chosen to provide the structure definition within the packets. As this is done on a per packet basis, the correct interpretation of packets is not impacted by packet losses.

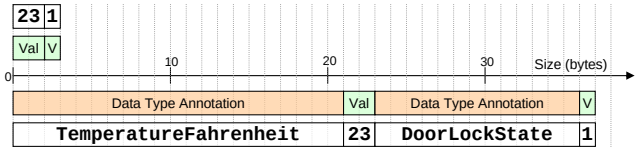


Fig. 1. Size of a packet without (top) and with (bottom) data type annotations

However the problem is that semantic data type annotations can lead to an increase of the packet size. In a simple example indicated in Fig. 1, only three bytes of payload values (`Val` and `V`) are transferred. The first field is 16 bits in size and carries a temperature reading, in the second field the state of a door lock is contained as boolean value. After the semantic data type annotation, the packet grows to 37 bytes in size. In this paper, we present an approach to reduce this overhead to a small fraction of the size shown, and highlight the benefits of dynamic packet composition. After presenting related work in Sec. II, we show a sample scenario benefitting from the use of annotations in Sec. III. We detail the use of semantic data type annotations in Sec. IV, and show in Sec. V how binary aliases can reduce the overhead introduced. We conclude this paper in Sec. VI, where we summarize our results and outline the next steps.

II. Related Work

The approach of introducing semantically annotated metadata in WSNs has been covered to some extent in related literature. The Open Geospatial Consortium (*OGC*) presents

an approach towards describing sensor devices in a semantic manner using the Sensor Model Language (*SensorML*) [7]. To enable the integration of such sensor systems into the semantic web, the Semantic Web Enablement (*SWE*) approach has been proposed in [8]. Both are however based on XML, and thus not sufficiently lightweight for application on embedded sensing devices. The Global Sensor Network (*GSN*) project [9] presents a middleware layer that abstracts all devices by *virtual sensors* and assigns semantic annotations. Inside the WSN, statically defined packets are used to transmit collected data.

Herzog et al. present the A3ME middleware in [10] with a special focus on the definition of sensor types and according messages. Their content representation is realized in a semantically annotated manner, with all pre-defined data types being stored in an ontology. Each time an unknown data type is present, it is indicated by an escape symbol followed by the complete description of the type. Embedded web servers present an emerging WSN application coping with variable packet payloads, enabled by applying TCP/IP to sensor networks [11]. In contrast to semantically defined packet payloads however, the application protocol defines how to decompose incoming messages and interpret their contents.

Maintaining all data types in an ontology represents the concept closest to our proposed use of dynamic payloads in WSNs. However, in contrast to a static data type ontology, we dedicatedly address possible dynamics in the network, i.e. new data types becoming present during runtime.

III. Illustration Scenario

The dynamic composition of packets is useful in many settings. In the remainder of this paper, let us e.g. envision a building surveillance sensor network, where each room is fitted with sensors configured to monitor a set of parameters. All nodes in the building form a routing tree and forward all their sensor readings to the root for processing and storage. An exemplary setup for one room is schematically shown in Fig. 2, integrating seven sensor nodes with twelve sensing modalities. While some of the sensors create continuous streams of data (such as noise level or brightness), others inherently generate events, e.g. indicating that a door or window has just been closed. Conventionally, all sensors would report their readings in a fixed interval, not taking the specific characteristics of the sensing devices into account.

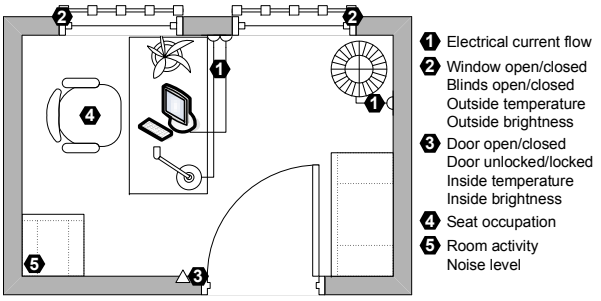


Fig. 2. Room monitoring scenario setup

Transferring each node's full set of sensor readings to the sink node, where it is stored for retrieval from third parties, in a statically defined packet leads to a complete update of the current building state, however it comes at the cost of packet payloads carrying the full set of sensor data. In contrast, confining packets to the relevant contents[1] through dynamic composition is a viable step towards preserving transmission energy. Also, when packets omit irrelevant fields from transmission, the resulting smaller payloads allow data fusion at intermediate nodes.

IV. Semantic Data Type Annotations

When packet payloads are no longer defined before the actual deployment phase, but instead composed during runtime depending on the availability of sensor data, the structural description of the packet payload must be present at the receiver to allow correct interpretation of the contents. Obviously, since providing a syntactic description of the data field type only (e.g. an unsigned integer of 16 bits length) would lead to ambiguities, semantic type annotations, such as `DoorLockState`, must be used supplementary. As semantic tags do neither imply their length nor the actual length of the data field, a length field and a separate syntactic type tag is used in combination with the semantic description.

The overall structure thus differs from the simplified form presented in Sec. I and is shown in Fig. 3. Every data field is now prefixed by both semantic and syntactic fields. First, the length of the semantic type tag is transmitted (entitled `L` in the figure), followed by the textual description of the semantic data type. Subsequently, the syntactic type of the following data is indicated (`T` in the figure, with types `S` indicating a 16 bit integer, while `B` refers to a boolean value). The syntactic type field is then followed by the actual sensor reading.

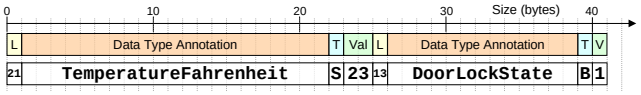


Fig. 3. Structure of a packet with syntactic and semantic data type annotations

As shown in the figure, we propose that metadata fields prefix the actual data field such that the received node can extract the type of data, the length of the corresponding field in the payload, and the value itself. After having provided a *meaning* to the value, all recipients who understand the given annotation tag can interpret the data accordingly. This enables data aggregation on nodes on the routing path, e.g. calculate the overall energy consumption in the entire building by multiplying `Voltage` and `Current` readings. Nodes to which the given data type is unknown can still forward the sensor data towards the sink. The length of the given payload field is defined by the syntactic description field, thus nodes can skip unknown types and proceed to the next field.

[1] As the process of determining relevant sensor data is beyond the scope of this paper, we assume that only significant changes to the sensor data (e.g. temperature shifts by at least one degree, or changes to the door lock state) necessitate the transmission to the building control server.

V. Consistent Assignment of Aliases

From the exemplary header structure shown in Fig. 3, it is clear that packet sizes are significantly increased when applying our proposed approach, as the semantic annotation of the data types requires the transmission of plaintext semantic data type descriptors. As long as readability by humans is intended, these verbose tags are well suited. However, in fully automated scenarios, valuable payload space can be saved by assigning aliases to the semantic data types. Instead of transmitting the plaintext value `TemperatureFahrenheit` with 21 characters, a field of a few bits in size can be used to represent the tag. In our design, we have used a field of one byte in size, which shares the same location as the length field. However, as the IEEE 802.15.4 standard [12] limits the packet size to 127 bytes, a maximum of seven bits can be required to represent the length of the semantic annotation. Using the remaining bit as an escape symbol, up to 128 data types may be present in the network. As a side effect, the constant length of the type field also obsoletes the corresponding length field.

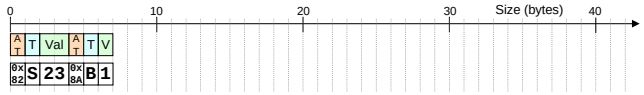


Fig. 4. Structure of a packet with semantic data type annotation aliases

Applying the to the previously shown packet structure, all semantic data type annotation fields are reduced to just a single byte. The resulting packet structure is visualized in Fig. 4 results, where a size reduction from 41 to 7 bytes can be seen (as compared to Fig. 3). Each entry is now only composed of the alias for the semantic type (`AT`, one byte), the syntactic field description (`T`, one byte), and the field itself (length inherently defined by the syntactic field description `T`).

In this presented approach, syntactic and semantic description elements are transferred separately from each other. If both are merged into a single descriptor field, another byte of payload size can be saved. Although feasible and beneficial in terms of size, we have deliberately decided to leave both fields uncoupled to allow other mechanisms to cater for efficient encoding of syntactic descriptors and data.

A. Definition of Aliases

The definition of aliases must take place in a coordinated order to avoid ambiguities. As the centralized sink node can provide the properties of transactional databases (i.e., atomicity, consistency, isolation, durability) best, we have imposed the tasks of assigning new aliases and storing a consistent mapping on this node.

To replace semantic data types by aliases on nodes, a local cache is implemented in all nodes. Whenever a sensor function returns data of a given semantic type, this local cache is checked for presence of an alias. In case an alias is known, the semantic type field in the outgoing packet is directly replaced by the shorthand notation. Otherwise, it is transmitted in plaintext, and a the creation of a new entry at the sink node is triggered, as described in the following section.

B. Adding Entries to the Dictionary

When binary shortcut forms for required data types are not known on the sensor nodes, they revert to the transmission of a plaintext semantic annotation, as shown in Fig. 3. Two possible situations may occur during the transport of the packet towards the sink:

1) An intermediate node has cached an according mapping between the plaintext annotation and the corresponding alias. In this case, the intermediate node replaces the field of the packet payload before relaying the data, aiming to minimize the size of the transmitted packet. In addition to forwarding a message towards the tree root, a notification message providing the corresponding mapping is also broadcast in the opposite direction (i.e., towards the origin of the packet), such that the sender as well as all nodes on the route may add the alias to their caches.
2) The data type has not yet been encountered in the network, and thus no mapping exists. In this case, the packet is forwarded to the sink node with plaintext data type annotation. There, a new entry is created and added to the mapping table, and the corresponding data type made known to the network through broadcasting.

In both cases, nodes should try to locally cache all annotation aliases for the data types they provide or consume, such that ideally, no plaintext annotations need to be transmitted after an initial setup phase. An excerpt of the mapping table generated in the exemplary room monitoring scenario depicted in Fig. 2 is shown in Table I.

C. Proactive Caching

In addition to storing mappings between for locally used semantic data type annotations only, nodes can also cache mappings for data types which are not used locally at all. While consuming additional RAM to store the mapping, this allows to resolve mappings closer to the node which has not yet stored the mapping for its data types (and thus sends the semantic data type annotation in plaintext). This way, a significant amount of traffic can be saved especially when the network is comprised of long routes. Caching mappings on intermediate nodes is possible, as binary aliases are only assigned by the central instance, i.e. the root node, so that no collisions can occur and a consistent state is guaranteed throughout the network runtime.

TABLE I
Aliases for Semantic Data Type Descriptors

Semantic Data Type	Alias
`RelativeHumidity`	`0x81`
`TemperatureFahrenheit`	`0x82`
`AccelerationX`	`0x83`
`WindowState`	`0x84`
`SwitchState`	`0x85`
`RelativeMotion`	`0x86`
`ElectricalVoltage`	`0x87`
`ElectricalCurrent`	`0x88`
`DoorState`	`0x89`
`DoorLockState`	`0x8A`

VI. Conclusion and Outlook

We have presented an approach to embed semantic data type annotations into packet payloads in WSNs. Opposed to static packet payload definitions, our solution allows to generate packets dynamically during runtime, and thus adapt to the characteristics of the attached sensor devices. Having shown that embedding semantic data types leads to significantly larger packet sizes, we have presented an approach to assign binary aliases to the data types, which are then used consistently throughout the network. Although the payload size is increased by two bytes per contained data field, the flexibility of dynamic packet composition allows to omit unchanged fields from transmission and enables more sophisticated in-network processing of data.

We dedicatedly focus on the efficient transfer of annotated data in wireless sensor networks and have therefore presented a mechanism to incorporate semantic elements into the network. It is essential to distinguish our design from related work where global ontologies describing sensor features are discussed. Any translation of the internally used data types to a global naming scheme is out of focus of our approach, but can be realized on a gateway device if necessary.

A. Outlook

As our next step, we target to complete the implementation of the presented mechanism for use on sensor nodes. We are planning to verify the effectiveness of our implementation on our TWiNS.KOM testbed [13], which integrates TelosB and SunSPOT devices. Special focus will hereby be put on scalable algorithms for the dissemination of aliases. In the long term, we are planning a real-world deployment of the resulting application on sensor nodes deployed in an office environment, like presented in Fig. 2, and target to investigate the achievable packet size and energy savings.

Acknowledgements

The authors would like to thank Parag S. Mogre and Stefan Schulte for the fruitful discussions and their contributions to this paper. This research has been supported by the German Federal Ministry of Education and Research (BMBF).

References

[1] J. Beutel, S. Gruber, A. Hasler, R. Lim, A. Meier, C. Plessl, I. Talzi, L. Thiele, C. Tschudin, M. Woehrle, and M. Yuecel, "PermaDAQ: A Scientific Instrument for Precision Sensing and Data Recovery in Environmental Extremes," in *Proceedings of the 8th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, 2009.

[2] K. Martinez, R. Ong, and J. Hart, "Glacsweb: A Sensor Network for Hostile Environments," in *Proceedings of the 1st IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, 2004.

[3] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, "Collection Tree Protocol," in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems (SenSys)*, 2009.

[4] B. Krishnamachari, D. Estrin, and S. Wicker, "The Impact of Data Aggregation in Wireless Sensor Networks," in *Proceedings of the International Workshop on Distributed Event-Based Systems (DEBS)*, 2002.

[5] T. Arici, B. Gedik, Y. Altunbasak, and L. Liu, "PINCO: A Pipelined In-Network COmpression Scheme for Data Collection in Wireless Sensor Networks," in *Proceedings of the 12th International Conference on Computer Communications and Networks (ICCCN)*, 2003.

[6] O. Dubuisson and P. Fouquart, *ASN.1: Communication Between Heterogeneous Systems*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001.

[7] M. Botts and A. Robin, "OpenGIS Sensor Model Language (SensorML) Implementation Specification," White Paper OGC 07-000, 2007.

[8] M. Botts, G. Percivall, C. Reed, and J. Davidson, "OGC Sensor Web Enablement: Overview and High Level Architecture," White Paper OGC 07-165, 2007.

[9] K. Aberer, M. Hauswirth, and A. Salehi, "The Global Sensor Networks middleware for efficient and flexible deployment and interconnection of sensor networks," EPFL, Tech. Rep., 2006. [Online]. Available: http://infoscience.epfl.ch/getfile.py?recid=83891

[10] A. Herzog, D. Jacobi, and A. Buchmann, "A3ME - An Agent-Based Middleware Approach for Mixed Mode Environments," in *Proceedings of the 2nd International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM)*, 2008.

[11] A. Dunkels, J. Alonso, and T. Voigt, "Making TCP/IP Viable for Wireless Sensor Networks," in *Work-in-Progress Session of the 1st European Workshop on Wireless Sensor Networks*, 2004.

[12] IEEE Std, "802.15.4 Part 15.4: Wireless medium access control (MAC) and Physical layer (PHY) specifications for low-rate wireless personal are networks (LR-WPANs)," 2006.

[13] A. Reinhardt, M. Kropff, M. Hollick, and R. Steinmetz, "Designing a Sensor Network Testbed for Smart Heterogeneous Applications," in *Proceedings of the 3rd IEEE International Workshop on Practical Issues in Building Sensor Network Applications (SenseApp)*, 2008.

Fine-grained Access Control Enabling Privacy Support in Wireless Sensor Networks

Delphine Christin*, Andreas Reinhardt†, Salil S. Kanhere‡, Matthias Hollick*
* Secure Mobile Networking Lab, Technische Universität Darmstadt, Darmstadt, Germany
† Multimedia Communications Lab, Technische Universität Darmstadt, Darmstadt, Germany
‡ School of Computer Science and Engineering, University of New South Wales, Sydney, Australia
{delphine.christin, matthias.hollick}@seemoo.tu-darmstadt.de, areinhardt@kom.tu-darmstadt.de, salilk@cse.unsw.edu.au

***Abstract*—The deployment of wireless sensor networks may endanger the privacy of people monitored on purpose or unintentionally. To enhance the control of the surveilled people over their privacy, we propose to introduce a fine-grained access control scheme on monitored data. Towards this end, we design an architecture relying on granular access control and filtering of relevant information, which can be easily integrated with existing wireless sensor networks.**

I. Introduction and Motivations

Wireless sensor networks (WSNs) are deployed to cover a wide range of scenarios ranging from environmental monitoring [1] to medical applications [2]. However, some deployments raise privacy concerns. The threats to privacy are directly identifiable in people-centric scenarios. However, environment-centric scenarios may also threaten the privacy of persons located in the monitored environment, as their current activity could be extracted from sensor data. To ensure that the deployed WSNs are well-received in their communities, respecting privacy is a condition *sine qua non*. Within the scope of this paper, we propose to involve the concerned people in the privacy decisions by introducing a fine-grained access control. The monitored persons are able to select potential data retrievers and pair them with an appropriate degree of granularity applied to the sensor data. Our approach takes into account the privacy preferences of each concerned individual and also optimizes the data collection process. In fact, some individuals may select less restricted privacy settings than the default settings, if they estimate that revealing this information does not affect their privacy.

Our contributions are as follows: We examine related work in Section II. We then analyze representative privacy-sensitive WSN scenarios in Section III in order to identify privacy issues and data retrievers involved. We present in Section IV our concept and the related architecture introducing fine-grained access control in WSNs. Then, we evaluate the proposed approach and highlight the advantages and limitations in Section V. Finally, in Section VI we summarize our discussion and provide an outlook on prospective future work.

II. Related Work

Ensuring *data privacy* and *context privacy* [3] in WSNs has attracted the interest of many researchers. While *data privacy* focuses on the privacy protection of the collected data and queries submitted to the WSN, *context privacy* tackles the protection of location and timing information related to the traffic streams. Several approaches (e.g. [4] and [5]) that use intermediate processing, such as data aggregation, have been introduced to ensure *data privacy* protection on a node-to-node basis. Mechanisms based on variations of the K-anonymity principle [6] have been studied in [7] to guarantee privacy of queries in WSNs. Solutions to ensure location privacy of data sources and base stations have been proposed in [8] and [9], respectively. Random delays [10] may also be introduced to ensure temporal privacy. It can be observed that existing work mainly focuses on the privacy protection of the WSN and that fined-grained access control mechanisms are missing. The AlarmNet project [2] is close to our concept and presents a flexible access control scheme depending on the monitored patient's context and potential health emergencies. The privacy preferences of the patient may be enforced and additional access authorizations may be delivered in case of emergency. This approach clearly introduces some flexibility in comparison with traditional role-based access control [11]; however, the authorized people access the same set of data with the same granularity. Moreover, its applicability is restricted to health care applications, while our scheme may be integrated into a wider range of WSN applications involving monitored people. To the best of our knowledge, we are the first to envisage a fine-grained access control to support privacy in wireless sensor networks. Our aim is to optimize the trade-off between privacy guarantees and information required by the application to fulfill its duties.

III. Selected Privacy-sensitive WSN Deployments

From the multi-dimensional WSN design space [12], we select representative privacy-sensitive scenarios: People-centric and environment-centric scenarios. By monitoring human beings in multiple contexts and environments, people-centric applications raise obvious privacy concerns. In comparison with people-centric scenarios, people are not the core study subjects in environment-centric deployments. However, their privacy may also be threatened, as their activity and location may be monitored in the background. To illustrate both categories of scenarios, we restrict our selection to three representative examples of deployments: Assisted living, monitoring employees, and smart homes. For each example, we identify different

sensed parameters and potential data consumers.

A. People-centric Scenarios

1) Assisted living: Wireless sensor networks are deployed in elderly peoples' homes to monitor their physiological parameters and activities. The deployment consists generally of two subnetworks: A Body Area Network (BAN) and a fixed network, as presented in [2]. Wearable sensors measuring ECG, blood pressure and sugar level may be part of the BAN, while presence detection, temperature and humidity sensors may compose the fixed network. The collected data can interest different groups of people including doctors, nurses, family members and other residents. However, to protect the privacy of the monitored people, these data should not be delivered with the same degree of granularity. Doctors require the entire set of the physiological parameters with the finest degree of granularity to be able to establish a diagnosis as precise as possible, while family members are only able to find out whether the monitored person is at home or in his room. Room temperature information might be made available to all people interested.

2) Monitoring Employees: The employees' current activity can be monitored by sensor nodes deployed in offices and workplaces in order to determine whether they can be interrupted, as presented in [13]. For example, cameras and accelerometers can be used to determine the employees' context. However, sensitive information is gathered and should be carefully disclosed to potential data requestors. Different degrees of granularity are introduced to allow superiors, colleagues, friends or family members to find out the current state of interruptibility of the person of interest. For example, the responses of the system indicates that the person is busy or reveal his activity with a high degree of precision depending on the trust level between the monitored employee and the requesting person.

B. Environment-centric Scenario

1) Smart Homes: Temperature, humidity, brightness, and contact sensors can be disseminated within habitations to measure the ambient conditions and control the Heating, Ventilating and Air Conditioning (HVAC) systems in order to optimize their energy consumption. Different groups of people can exploit these measurements: The residents, the employees of the surveillance company and the firefighters. The residents can consult the sensor data to verify if the measured values correspond to their preferences or adjust them to their needs, while the surveillance company and the firefighters may consult the data to detect intrusions and fires respectively. Authorizing the access to the sensor data with the highest degree of granularity to the residents does not raise any privacy concerns. On the contrary, the privacy of the habitants may be endangered if the surveillance company and the firemen have access to these data and are able to extract the current activities of the residents. For example, if the light is on and the humidity degree increases in the bathroom, it can be easily deduced that somebody is taking a shower. To protect the privacy of the habitants, the sensor data should be delivered to the third parties with a coarser degree of granularity. However, the selected degree of granularity should be sufficient to allow them to detect abnormal events.

The examination of the previous scenarios has showed that distinct categories of people have interests in accessing the sensed data and demonstrated the necessity to introduce different degrees of granularity to ensure privacy. However, proposing a general mapping between categories of data consumers and degrees of granularity is made impossible by the personal nature of the privacy conception.

IV. Concept and Architecture

Based on the observations made in the previous section, we propose an approach to introduce fine-grained access controls supporting privacy in WSNs. We first introduce briefly our concept and we then provide a detailed description of the proposed architecture and the related components.

A. Concept Overview

Guaranteeing privacy in WSNs is a key factor to allow their acceptance by the public. In fact, without any privacy protection mechanism, their deployment may be refused or the monitored people may deactivate the sensing function. In this case, the sensor data collected by the application may be insufficient to deliver reliable results and fulfill its primary function. To avoid such pitfalls, we propose to optimize the trade-off between guaranteeing the protection of privacy and fulfilling the application needs by introducing a fine-grained access control mechanism. The concerned people are able to select the persons authorized to access their data and attribute them different privacy degrees depending on the nature of the data, their privacy preferences and their personal relationships. The mechanism of data delivery is not on a binary basis, available or unavailable, but becomes granular with multiple degrees. Consequently, more parties can benefit from the data without threatening the privacy of the monitored subjects.

B. Architecture

Our concept is based on a two-tier architecture including a WSN and a privacy-enhanced base station, as illustrated in Fig. 1. The base station supports interaction with the monitored persons, as well as potential data consumers.

1) Wireless Sensor Network: The WSN is composed of homogeneous or heterogeneous platforms. We assume that the sensor data are annotated with timestamps and that the base station can identify the type of sensor data. The data are transferred to the base station via traditional routing protocols adapted to characteristics of the deployment. The communication between the sensors and the base station is assumed to be adequately protected against external adversaries by well-established cryptographic techniques e.g. encryption and authentication. The integration of complex techniques using different encryption keys depending on the required level of privacy, such as presented in [14] may also be envisaged. Once the data reaches the base station, their characteristics

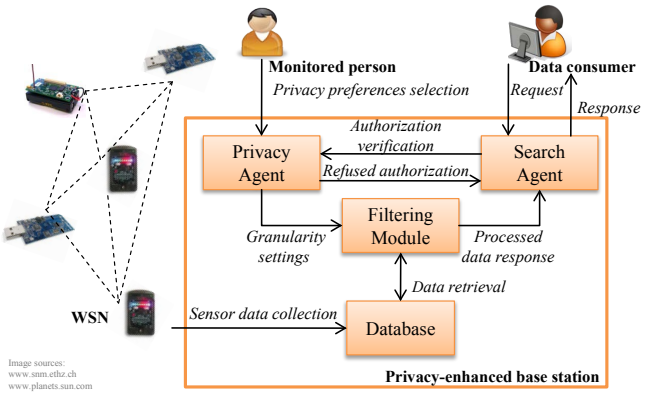


Fig. 1. Proposed Architecture

are extracted (e.g. timestamps, data type) and they are then stored in the database. The database is secured and protected against attackers by means of classical mechanisms.

2) Privacy-enhanced Base Station: It comprises four main components: A search agent, a privacy agent, a filtering module, and a database. The functionalities of these components are successively described in detail in the following paragraphs.

- *Search Agent*: Data consumers can login and search for data through a web interface. They can define different search parameters including e.g. type of data and collection time. These parameters are included into a data request and transmitted to the search agent. The search agent processes it and delegates the authorization verification to the privacy agent. If the data consumer is identified as unauthorized by the privacy agent, the search agent displays an "access denied" message. Otherwise the search agent displays the response provided by the system, once the full processing cycle is achieved.
- *Privacy Agent*: Each monitored person can define his/her privacy preferences through a dedicated interface. The privacy preferences include a list of authorized people mapped with a particular degree of granularity for each type of sensed data. The privacy agent stores, manages and maintains these privacy preferences. When a data request for accessing a particular data is received by the search agent, it forwards the same to the privacy agent. The privacy agent verifies the access authorization set by the concerned person(s). In case of diverging privacy settings between multiple monitored persons, the privacy agent selects the strictest preferences in order to ensure the maximal privacy guarantees for each person. For example, Alice and Bob are monitored in the same room and Alice authorized Carol to access her data with a high degree of granularity, while Bob did not mention Carol in his list of authorized people (or indicate a lower degree of granularity than Alice). Carol will thus not be authorized to access the requested data (or only with the lowest degree of granularity set by Bob). When the authorization is checked and the degree of granularity determined, the privacy agent transmits a data request including the corresponding data and granularity settings to the filtering component.
- *Filtering Module*: The filtering module is composed of different submodules. Each submodule supports a unique type of data and is able to extract different degrees of granularity from the sensor data. The amount and nature of the submodules is influenced by the WSN application and the deployed sensors. Moreover, the complexity of the processing depends on the nature of the sensor data. For example, extracting distinct degrees of granularity from temperature data requires simpler processing than from captured sounds. Once the filtering module receives the data request from the privacy agent, it retrieves the data from the database and conveys it to the dedicated submodule by indicating the desired degree of granularity. As soon as the processing is achieved, the resulting data are transmitted at the appropriate level of granularity to the search agent.

V. Evaluation and Discussion

To complete the presentation of our approach, we conduct a preliminary evaluation of our concept and highlight the key advantages and limitations. The discussions focus on two main dimensions: Technical and human aspects.

From a technical point of view, our approach can be integrated into a wide range of WSN applications without requiring major modifications of the WSN deployment. The protocols and mechanisms applied within the WSN remain unchanged. However, sensor registration or data annotations are required to identify the type of data sensed and to apply the correct processing in the filtering module. If no scheme to identify the data type is supported in the existing WSN, its introduction may cause additional overhead and complexity. The design and implementation of the base station would require some modifications in order to introduce the agents discussed in Section IV. The functionality of the privacy and search agent would not differ to a large extent from one application to another. Only the available types of sensor data have to be adapted in the selection of the privacy settings and the search process. Therefore, a generic design can be developed and easily reused by adapting the data types to the application requirements. Similarly, a library of submodules for the filtering module covering a large range of sensor data to process could be designed and implemented. By maintaining their structure modular and loosely coupled, such submodules could be integrated easily. Nonetheless, the dependencies of the architecture on the sensor data types limit the flexibility of the WSN. For each new sensor type introduced within the WSN, an additional submodule has to be added into the filtering module.

Considering the human dimension, our concept introduces a novel perspective for the monitored people by involving them in the privacy decisions. Instead of employing static and generic privacy protection mechanisms, the concerned people can tune the privacy settings (authorized parties and associated granular data) according to their preferences. In

comparison with existing work, our approach allows to take into account different privacy conceptions of individuals that cause potential conflicts between the concerned parties. Multiple monitored people may have conflicting privacy settings, as described in detail in Section IV.B. In this case, the strictest privacy settings among the conflicting settings are adopted to provide the maximal privacy protection. The privacy of the monitored people is therefore not jeopardized. Additionally, the selected privacy settings may be stricter than the minimal degree of granularity required by the application. Depending on the application scenario, the person can choose to modify her settings or leave the deployment, or the application enforces the privacy preferences in case of emergency. In the latter case, the notion of emergency should be clearly defined and the monitored people well-informed of such potential enforcements. Furthermore, we assume that each monitored person is able to define her privacy preferences. This assumption requires that the monitored people are known and identifiable. The applicability of our concept may therefore be limited in particular deployments, such as in public areas e.g. train stations, where people are entering and leaving the monitored location frequently. However, the crowd present in such locations protects to some extend the privacy of the monitored people, as personal data are difficult to extract.

VI. Conclusions and Outlook

In this paper, we have analyzed representative privacy-sensitive WSN scenarios and highlighted that privacy can be partially supported by allowing data access with different degrees of granularity depending on the nature of the relationships between the monitored people and the data consumers. We have then described our concept to introduce fine-grained access controls in WSNs and have proposed an architecture supporting the deployment of our approach in real scenarios. We have detailed the functionality of each component of the proposed architecture. Finally, we have conducted a preliminary evaluation of our concept to highlight the advantages and limitations of our approach by focusing on the technical and human perspectives.

In conclusion, our approach provides a generic solution adapted to a wide range of privacy-sensitive WSN scenarios demanding only limited adaptations to the application characteristics. By employing our concept within WSN deployments, the trade-off between privacy respect and application needs is optimized. Instead of disabling partially or completely the sensing capability of the deployment rendering the application inoperative to protect their privacy, the monitored people can finely define access authorizations including authorized parties and corresponding degrees of granularity. Moreover, the direct involvement of the monitored people in the privacy decisions and privacy setting selection may improve their acceptance of WSN deployment, as they have a direct influence on the mechanisms employed to protect their privacy.

A. Outlook

As future work, we plan to provide a proof-of-concept of our approach to complete the conducted conceptual evaluation. The implementation will include an extensive library of submodules for the previously defined filtering in order to provide off-the-shelf components that may be combined effortlessly in multiple combinations depending on the sensor data types. Additionally, we will pay particular attention to the design of interfaces used by the monitored people to select their privacy preferences. In fact, the selection process may rapidly become cumbersome for the user, if it includes the personalization of numerous parameters. Usability and simplicity will therefore be particularly considered.

In a second phase, we foresee to expand this approach to other platforms and build hybrid networks composed of traditional sensor platforms (e.g. Mica2, SunSPOT) and personal end devices (e.g mobile phones). Such mixed networks will benefit from the sensors embedded in the personal end devices providing mobility patterns, as well as convenient interfaces to support direct interactions between the monitored people and the network infrastructure.

Acknowledgment

This work was supported by CASED (www.cased.de).

References

[1] K.Martinez et al., "Glacsweb: A Sensor Network for Hostile Environments," in *Proc. of the Sensor and Ad Hoc Communications and Networks Conference (SECON)*, 2004.

[2] A. Wood et al., "Context-Aware Wireless Sensor Networks for Assisted-Living and Residential Monitoring," *IEEE Network*, 2008.

[3] N. Li et al., "Privacy Preservation in Wireless Sensor Networks: A State-of-the-art Survey," *Ad Hoc Networks*, vol. 7, no. 8, 2009.

[4] W. He et al., "PDA: Privacy-Preserving Data Aggregation in Wireless Sensor Networks," in *Proc. of the IEEE International Conference on Computer Communications (INFOCOM)*, 2007.

[5] W. Zhang et al., "GP^2S: Generic Privacy-Preservation Solutions for Approximate Aggregation of Sensor Data," in *Proc. of the IEEE International Conference on Pervasive Computing and Communications (PerCom)*, 2008.

[6] L. Sweeney, "k-anonymity: A Model for Protecting Privacy," *International Journal of Uncertainty Fuzziness and Knowledge Based Systems*, vol. 10, no. 5, 2002.

[7] B. Carbunar et al., "Query Privacy in Wireless Sensor Networks," *ACM Transactions on Sensor Networks*, vol. 6, no. 2, 2010.

[8] P. Kamat et al., "Enhancing Source-Location Privacy in Sensor Network Routing," in *Proc. of the IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2005.

[9] J. Deng et al., "Decorrelating Wireless Sensor Network Traffic to Inhibit Traffic Analysis Attacks," *Pervasive and Mobile Computing*, vol. 2, no. 2, 2006.

[10] P. Kamat et al., "Temporal Privacy in Wireless Sensor Networks," in *Proc. of the International Conference on Distributed Computing Systems (ICDCS)*, 2007.

[11] D. Ferraiolo et al., *Role-based Access Control*. Artech House Publishers, 2003.

[12] K. Römer et al., "The Design Space of Wireless Sensor Networks," *IEEE Wireless Communications*, vol. 11, no. 6, 2004.

[13] A. Reinhardt et al., "Towards Seamless Binding of Context-aware Services to Ubiquitous Information Sources," in *Proc. of the International Conference on Complex, Intelligent and Software Intensive Systems (CISIS)*, 2010.

[14] M. Shao et al., "pDCS: Security and Privacy Support for Data-Centric Sensor Networks," in *Proc. of the IEEE International Conference on Computer Communications (INFOCOM)*, 2007.

Senkung der Versicherungsprämien bei Überlandtransporten mittels geeigneter Smart Object Technologien

Hauke Traulsen, Christopher Kaffenberger, Alexander Pflaum
Geschäftsfeld Technologien
Zentrum für Intelligente Objekte ZIO
Fraunhofer-Arbeitsgruppe für Supply Chain Services SCS
Fraunhofer-Institut für Integrierte Schaltungen IIS
Dr.-Mack-Straße 81
90762 Fürth
{vorname.nachname}@scs.fraunhofer.de

***Abstract*— Um das Risiko auf Überlandtransporten abzuschätzen, werden von Versicherungen Risikoanalysen durchgeführt. Auf dieser Basis werden anschließend Prämien beziehungsweise Beiträge ermittelt. Durch den Einsatz von Technologien intelligenter Objekte, insbesondere durch drahtlose Sensornetzwerke, könnten Risiken zukünftig minimiert und Versicherungskosten für Logistikunternehmen gesenkt werden. Die Abbildung von risikomindernden Faktoren, welche von den Versicherungen festgelegt werden, auf Funktionen technologischer Lösungen, welche überwiegend von den Herstellern geprägt sind, ist dabei eine der zentralen Herausforderungen. Am Zentrum für Intelligente Objekte [5] wird genau diese Herausforderung adressiert. Ziel ist, mittelfristig Lösungen zu schaffen, die sowohl die Anforderungen der Versicherer als auch die der Technologieanwender erfüllen.**

I. EINLEITUNG

Versicherungen ermitteln die Höhe von Prämien mit Hilfe teilweise komplexer Risikoanalysen. Transportversicherungen werden in der Logistik beispielsweise abgeschlossen, um den Ausfall einer Produktionsanlage, entstanden durch ungewünschte Vorfälle auf der Transportstrecke zwischen Zulieferern und OEMs, finanziell abzusichern. So kann ein Gut während des Transports durch aktives Fremdeinwirken beschädigt oder gestohlen werden. Die Ware fehlt am Zielort und verursacht zusätzliche Kosten. Teure Verzögerungen in globalen Produktionsnetzen können dadurch entstehen, dass ein Container in Fernost versehentlich nicht oder auf das falsche Schiff verladen wird und dass die Zulieferteile am Zielort in Europa eben nicht Just-in-Time zur Verfügung stehen. Diese beiden Beispiele stehen für eine lange Liste von Problemen, die während des Transports auftauchen können. Es sind Lösungen gefragt, mit denen diese Probleme reduziert oder vollständig gelöst werden. Im Falle eines nachträglich erkannten Schadens muss zumindest eindeutig nachweisbar sein, welcher Akteur innerhalb der logistischen Kette der Verursacher ist. Im Idealfall generiert die technische Lösung Informationen, welche ein frühzeitiges Erkennen von Problemen und das präventive Abwenden von finanziellen Schäden erlauben. Durch eine solche Lösung gewinnt das Logistiksystem an Zuverlässigkeit, Robustheit und „Integrität“.

Technologien intelligenter Objekte wie zum Beispiel RFID, drahtlose Sensornetzwerke und Lokalisierungssysteme können bei der Lösung des eben geschilderten Problems helfen. Mit Hilfe eines Smart Object-basierten „Supply Chain Integrity Monitoring Systems“ könnte tatsächlich ein großer Teil der heute noch vorhandenen Probleme gelöst werden. Design und Umsetzung eines solchen Systems erfordern allerdings einen intensiven Diskurs zwischen Versicherern, Anwendern und Technologieanbietern. Im Rahmen dieses Diskurses müssten die zu adressierenden Probleme und Integritätsverletzungen adressiert, Anforderungen an die Technik herausgearbeitet und entsprechende Lösungen spezifiziert sowie aus ökonomischer Sicht bewertet werden. Im Kern ist von den Beteiligten die Frage zu beantworten, in wie weit sich mit einem Smart Object-basierten Informationsdienst Risiken in logistischen Transportketten tatsächlich reduzieren lassen können und wie sich die Geschäftsmodelle der Versicherer verändern müssen. Voraussetzung ist dabei immer, dass alle beteiligten Akteure gewinnen. Die eben beschriebene Aufgabe wird nur dann bewältigt werden können, wenn hinsichtlich des erforderlichen Smart Object-basierten Informationssystems und des dahinter liegenden Dienstes klare Vorstellungen existieren, an denen sich die betroffenen Akteure orientieren können.

Im vorliegenden Papier soll deswegen die Vision eines technischen Systems zur Erkennung möglicher Verletzungen der Integrität globaler Wertschöpfungsketten und logistischer Systeme skizziert werden. Es soll Versicherer, Dienstleister aus dem Bereich der Logistik, produzierende Unternehmen und Technologieanbieter für die oben genannte Problemstellung sensibilisieren. Im Anschluss wird eine Vorgehensweise für die betriebswirtschaftliche Bewertung des Systems vorgeschlagen. Sind Kosten und Nutzen des Systems bekannt sowie den einzelnen Akteuren im Gesamtsystem zugeordnet, lässt sich unter Umständen ein Geschäftsmodell entwickeln, welches den Versicherern die Reduktion der Versicherungsprämien erlaubt. Die detaillierte Betrachtung des Nutzens und der Systemkosten ist noch nicht Bestandteil des vorliegenden Papiers. Sie kann im Grunde nur im Nachgang unter Beteiligung der bereits oben mehrfach genannten Akteure erfolgen und muss Gegenstand einer umfassenderen Untersuchung sein.

II. Integritätsschutz Durch Smart Object Technologien im Projekt Aletheia

Die weiteren Überlegungen zugrunde liegende technische Vision kann sehr gut anhand des vom Bundesministerium für Bildung geförderten Leitprojekts Aletheia [1] erklärt werden. Hier arbeiten das Fraunhofer Institut für integrierte Schaltungen [2], das DHL Innovation Center, Giesecke und Devrient sowie Euro-Log an einem Smart Object-basierten Informationssystem zur Überwachung der Integrität von Versorgungsketten. Der Fokus liegt auf europäischen Landtransporten. Abbildung 1 zeigt das hier zu entwickelnde System im Überblick. Auf Produkten, an Verpackungen sowie an Seecontainern werden Sensorknoten angebracht, welche hierarchisch mit einander vernetzt sind. Funktionalität und Leistungsfähigkeit der Knoten auf den verschiedenen Hierarchieebenen sind unterschiedlich.

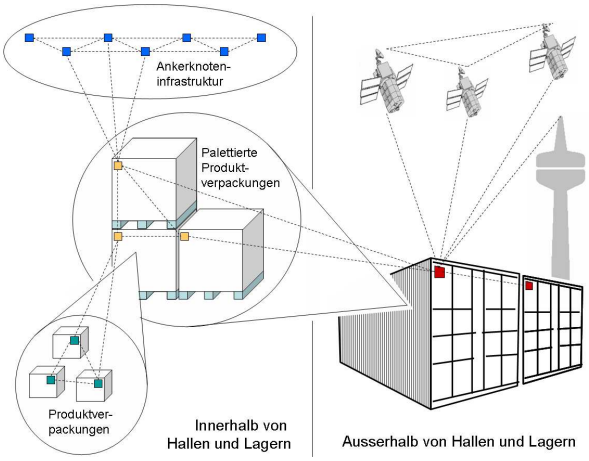


Abbildung 1: Hierarchisches Sensornetzwerk aus Aletheia

Innerhalb von Gebäuden kann eine eigene Infrastruktur aus fest angebrachten Ankerknoten, außerhalb von Gebäuden das satellitenbasierte Ortungssystem GPS für die Lokalisierung einzelner Einheiten beziehungsweise Objekte genutzt werden. Die Nutzenpotenziale, die durch das System zukünftig realisiert werden könnten, sind vielfältig. Komplexe Knoten mit GSM-Anbindung einerseits und zusätzlicher Sensorik andererseits an den Containern sorgen für eine durchgängige Überwachung auf der gesamten Transportstrecke. Jederzeit kann das unberechtigte Öffnen eines Containers durch Dritte sicher erkannt werden. Die unmittelbare Übermittlung einer Nachricht in Echtzeit an eine Überwachungszentrale erlaubt schnelles Eingreifen durch Sicherheitskräfte vor Ort. Schwund kann vermieden, entsprechende finanzielle Verluste können zukünftig minimiert werden. Weiterer Nutzen wird in Zukunft auch auf der Ebene der Paletten und der Produkte entstehen. Miniaturisierte und miteinander vernetzte Sensorknoten lassen sowohl Paletten als auch Produktverpackungen zu intelligenten Objekten werden. Diese Objekte überwachen die drahtlose Kommunikationsverbindung zu den jeweiligen Nachbarn und können so feststellen, ob sie selbst oder eines der Nachbarobjekte aus dem Verbund „Palette“ unberechtigter Weise entfernt werden. Auch hier kann möglicher Schwund erkannt und in vielen Fällen zukünftig wohl auch vermieden werden. Die intelligenten Objekte der Zukunft werden des Weiteren in der Lage sein, sensorisch Umgebungsparameter und damit auch den eigenen Zustand zu erfassen. Auf eine mögliche Abweichung von Normwerten kann direkt reagiert werden. Wird beispielsweise ein für einen bestimmten Kontext vorgegebener Temperaturwert beim Transport von gekühlt zu transportierenden Produkten erkannt, lässt sich wieder über eine Überwachungszentrale schnell die nötige Maßnahme ergreifen.

Auf allen drei Ebenen werden auf den jeweiligen Objekttyp abgestimmte Regeln durch eine direkt auf den entsprechenden Knoten realisierte Rule Engine überprüft. Die Regeln können hierbei durchaus komplex werden und eigene sowie von Nachbarknoten generierte Messwerte beziehungsweise Events miteinander verknüpfen. Für den Fall, dass von einem Knoten im System ein Alarm generiert wird, wird dieser über sämtliche Ebenen hinweg bis zu einer Überwachungszentrale weiter vermittelt und dort verarbeitet. Ist die Verbindung zur nächsten Ebene im System an der einen oder anderen Stelle temporär nicht verfügbar, wird die Alarmnachricht auf der letzten, noch erreichbaren Ebene im System zwischengespeichert und zu einem späteren Zeitpunkt erneut ein Versuch unternommen, diese weiter zu versenden. Jede auf einem Knoten implementierte Regel kann in der Überwachungszentrale über das Netzwerk konfiguriert werden. Regeln lassen sich zum Beispiel aktivieren, deaktivieren oder hinsichtlich ihrer Überwachungsfrequenz verändern. Neue Regeln können jederzeit drahtlos und während des Betriebs des Systems aufgespielt werden, was ein sehr komfortables Instrumentarium darstellt, um dynamisch die Prozesslogik des Knotens auf Anwendungsebene zu verändern.

Die nachfolgende Abbildung zeigt die Systemarchitektur eines in Aletheia entstandenen Demonstrators zur Umsetzung der weiter oben beschriebenen Vision.

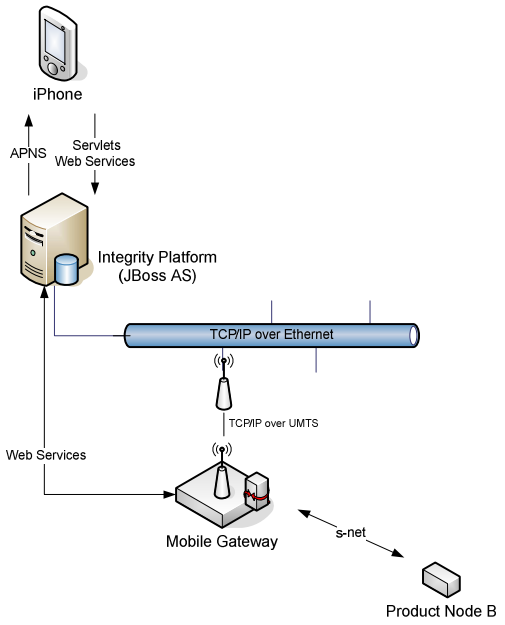


Abbildung 2: Auszug aus der Systemarchitektur des Aletheia-Demonstrators

Das „mobile Gateway“ wird hier auf einer Europalette als Palettenknoten verwendet. Der mit dem Gateway drahtlos vernetzte Sensorknoten „Product Node B“ wird als Produktknoten an einem verpackten hochwertigen Konsumgut angebracht. In der praktischen Umsetzung würden sämtliche Verpackungen auf der Palette mit einem Produktknoten ausgestattet sein. Das drahtlose Sensornetz auf Produktebene basiert auf den S3-Knoten [6] des Fraunhofer Instituts für Integrierte Schaltungen IIS, welche mit dem Protokoll s-net™ [3] über ein extrem stromsparendes Kommunikationsprotokoll verfügen. Das mobile Gateway, welches ebenfalls von Fraunhofer entwickelt wurde, besteht aus einem modifizierten, auf Energieeffizienz optimierten mobilen Embedded Linux PC mit verschiedenen Kommunikationsschnittstellen. Für den Demonstrator wird neben s-net™ UMTS/GSM für die drahtlose Anbindung an eine Überwachungszentrale genutzt. Neben der Weiterleitung von Nachrichten kommt dem Gerät die Aufgabe der Anreicherung von Nachrichten aus dem Sensornetz bspw. mit GPS-Koordinaten und globalen Zeitstempeln zu. Die Integrationsplattform, welche die eigentliche Supply Chain Integrity-Anwendung mit dem mobile Gateway verbinden soll, wurde auf einem JBoss Application Server aufgesetzt. Die Wahl fiel auf diese Open Source Lösung, weil hier Web Services (speziell WS-Eventing) besonders gut unterstützt werden können. Um unterschiedliche Akteure beziehungsweise Personen mobil und komfortabel mit Informationen aus dem System versorgen zu können, wurde für gängige mobile Plattformen eine passende Anwendung bereitgestellt. Abbildung 3 zeigt die einzelnen Bestandteile des Demonstrators.

Abbildung 3: Komponenten des Aletheia-Demonstrators

Als mobile Plattform wurde in diesem Fall ein iPhone verwendet. Hinter dem Mobiltelefon ist eine Version des Mobile Gateway zu sehen, rechts davon ein Produktknoten auf S3Tag Basis, jeweils in einem Schutzgehäuse. Das hier extern zu erkennende zusätzliche Batteriepaket dient Testzwecken. In der operativen Anwendung wird die Energieversorgung auf Basis geeigneter Batterien in das Gehäuse verlegt.

III. RISIKOMINIMIERUNG DURCH SENSORNETZWERKE

Der letzte Abschnitt hat gezeigt, dass durch den Einsatz von Sensornetzwerken Gefahren in unternehmensübergreifenden Transportketten erkannt und durch Einleitung entsprechender Gegenmaßnahmen Wertverluste gemindert, unter Umständen sogar verhindert werden können. Geklärt werden muss an dieser Stelle noch, mit welcher Argumentation Versicherungen dazu gebracht werden können, Versicherungsprämien dann auch wirklich zu senken. Versicherer verfügen intern über sehr detaillierte und komplexe Rechenmodelle, mit denen sie die möglichen Verluste quantifizieren können, welche aufgrund heute vorhandener Bedrohungen und Risiken ohne Einsatz von Technologien intelligenter Objekte entlang des Transportwegs mit einer gewissen Wahrscheinlichkeit zu erwarten sind. Auf dieser Basis werden die entsprechenden Versicherungsbeiträge ermittelt. Für Standardsicherheitslösungen im IT Bereich existieren bereits mehr oder minder genormte Anforderungen, die eine Versicherung an einen Kunden stellt, wenn dieser eine Minderung des Versicherungsbeitrags erreichen möchte. Anders verhält es sich bei neuen Lösungen auf Basis von Smart Object-Techniken wie Radiofrequenzidentifikation, drahtlosen Sensornetzwerken und Lokalisierungssystemen. Hier handelt es sich heute noch um prinzipiell innovative und komplexe Spezialanfertigungen, für die sowohl die Systemkosten als auch die zu erwartenden Nutzenpotenziale nur sehr schwer zu ermitteln sind. Hier fehlt den Versicherern neben detailliertem Wissen die Technologie betreffend auch das geeignete Verfahren für die Beurteilung des Potenzials, Risiken entlang der Transport- oder Versorgungskette zu minimieren.

Basis für ein erstes pragmatisches Verfahren kann die aus der Produktentwicklung bekannte Fehlermöglichkeits- und Einflussanalyse FMEA sein. In einem ersten Schritt werden gemeinsam mit betroffenen Anwendern und Versicherern möglich Risiken identifiziert, die innerhalb eines gegebenen logistischen Systems beziehungsweise einer Transportkette auftreten können. Diese lassen sich im Anschluss hinsichtlich der Wahrscheinlichkeit des Auftretens, der Schadenswirkung sowie der Entdeckungswahrscheinlichkeit beurteilen und auf dieser Basis bezüglich ihrer Bedeutung klassifizieren und ordnen. Jedes Risiko lässt sich durch ein Funktionsbündel, welches durch ein Smart Object-basiertes Informationssystem zur Gewährleistung der Integrität von Versorgungsketten zur Verfügung gestellt wird, reduzieren. Die Leistungsfähigkeit der verwendeten Produkte beziehungsweise Basistechnologien bestimmt das Ausmaß der Risikoreduktion. Bei genügend detaillierter Betrachtung lässt sich nicht nur dieses Ausmaß detailliert beschreiben. Es wird plötzlich möglich, Wertverluste beziehungsweise Benefits quantitativ abzuschätzen, welche auf Seiten des Versicherers entstehen. Auf der anderen Seite können durch die betroffenen Parteien klare funktionale und nicht-funktionale Anforderungen an die technologische Lösung abgeleitet und gemeinsam mit Technologieanbietern einfach in Systemkosten umgesetzt werden. Vorgehensmodelle des Service Engineerings wie zum Beispiel das am Fraunhofer Zentrum für intelligente Objekte ZIO in den letzten Jahren entwickelte Nürnberger Service Engineering Binokular NSEB sowie unterstützende Methoden und Werkzeuge erlauben die Konstruktion eines entsprechenden Sicherheitsdienstes und die Entwicklung eines entsprechenden Geschäftsmodells. Letztlich liegen mit diesem Schritt alle erforderlichen Informationen vor, die für eine Umsetzungsentscheidung durch das Management der betroffenen Unternehmen ermöglichen. Ob der Versicherer, der Technologieanbieter, der Logistiker oder ein IT-Integrator den entwickelten Sicherheitsdienst

anbietet, ist zunächst unerheblich. Diese Frage kann bzw. muss im Einzelfall entschieden werden.

IV. Zusammenfassung und Ausblick

Das vorliegende Papier ging von der Frage aus, wie die Technologien intelligenter Objekte genutzt werden können, Risiken in Versorgungsketten zu minimieren und Versicherer dazu zu veranlassen, Versicherungsbeiträge zu verringern. Anhand des Projektes Aletheia wurde die Vision für das hierfür erforderliche Informationssystem aufgezeigt. Deutlich wurde hier, dass drahtlose Sensornetzwerke, wie sie heute bereits in vielen anderen Anwendungen genutzt bzw. diskutiert werden, eine gute technische Basis für eine Problemlösung darstellen. Anhand einer Reihe von Beispielen wurden erläutert, welche Nutzenpotenziale sich hinter der Technologie verbergen und wie die heute in globalen logistischen Systemen vorhandenen Bedrohungen abgebaut werden können. Mittels eines ersten Demonstrators wurde gezeigt, wie die Vision mit Hilfe von heute schon vorhandenen Produkten und neuen technischen Entwicklungen umgesetzt werden kann. Das Ziel war hier, Versicherer, Technologieanbieter und Anwender für die zugrunde liegende Problematik und die möglichen technischen Lösungsansätze zu sensibilisieren.

Unabhängig hiervon wurde aber auch deutlich, dass eine pragmatische und trotzdem wissenschaftlich fundierte Methode für eine praxisnahe betriebswirtschaftliche Bewertung der Risikominimierungspotenziale und der entsprechenden Kosten eines Gesamtsystems noch nicht zur Verfügung steht. Hier wurde eine Vorgehensweise vorgeschlagen, welche einerseits auf der bekannten Fehlermöglichkeits- und Einflussanalyse, andererseits auf neuen Vorgehensmodellen und Methoden des Service Engineerings basiert. In diesem Zusammenhang wurde besonders auf das Nürnberger Service Engineering Binokular hingewiesen. Der nächste Schritt wäre dementsprechend die Anwendung dieser Vorgehensweise in der Praxis. Wesentlich für den Erfolg ist an dieser Stelle, dass Versicherer, Anwender und Technologieanbieter eng zusammen arbeiten und die Erarbeitung einer Lösung gemeinschaftlich treiben. In wie weit eine solche Kooperation tatsächlich möglich ist, muss die Zukunft zeigen.

Zu bedenken ist hierbei noch, dass ein Versicherer ohne einen klaren Nachweis des risikominimierenden Effekts Prämien im Vorhinein nicht senken wird. Es wird nötig sein, eine erste Lösung zu entwickeln und über eine gewisse Zeit zu betreiben. Nur dann, wenn der Effekt tatsächlich beobachtbar und nachgewiesen ist, wird es zu einer Absenkung der Beiträge kommen. Dieser Punkt muss bei der Entwicklung einer Lösung und bei deren Einbettung in ein Service- bzw. Geschäftsmodell dringend beachtet werden. Ein Verfahren zur Abschätzung der Risikoreduzierung kann deshalb zunächst wohl nur als unverbindliche Kommunikationsbasis zwischen Versicherern, Anwendern und Technologieanbietern dienen. Eine solche gemeinsame Kommunikationsbasis zu besitzen ist trotzdem unerlässlich, um es Technologieanbietern zu ermöglichen, das Potenzial ihrer entwickelten Lösungen abzuschätzen und Anwender mit dem Segen der Versicherer vom Einsatz der Technologie zu überzeugen.

References

[1] Aletheia - Semantische Föderation umfassender Produktinformationen, Online im Internet: www.aletheia-projekt.de [Stand 05.07.2010]

[2] Fraunhofer-Institut für Integrierte Schaltungen IIS, Online im Internet: www.iis.fraunhofer.de [Stand 05.07.2010]

[3] S-net™ Die Technologie des Fraunhofer IIS für extrem energiesparende, drahtlose Sensornetze, Online im Internet: www.iis.fraunhofer.de/bf/ec/dk/sn/index.jsp [Stand 05.07.2010]

[4] Kommunikationsnetze, Online im Internet: www.iis.fraunhofer.de/abt/kom/index.jsp [Stand 05.07.2010]

[5] Zentrum für Intelligente Objekte ZIO, Online im Internet: www.zio.fraunhofer.de [Stand 05.07.2010]

[6] S3TAG Funkmodul für drahtlose Datenübertragung, Online im Internet: http://www.iis.fraunhofer.de/fhg/Images/KOM_SSNW_MHW-1000548_0102_0812_tcm97-137075.pdf [Stand 06.07.2010]

Wireless Energy Transmission for Implantable Wireless Sensor Nodes

Tristan Bremer, Maike Vollmer
Comprehensive Hearing Center
University Hospital Wuerzburg
Email: Bremer_T@klinik.uni-wuerzburg.de / Vollmer_M@klinik.uni-wuerzburg.de

***Abstract*—This report describes a wireless powering method for implantable wireless sensor nodes. New Radio-Frequency Identification (RFID) integrated circuits provide energy to implanted biosensors and permit charging a small lithium accumulator. This method allows the long term use of fully implanted biosensors and, thus, provides a new method for physiological recordings and behavioral research. A comparison with previously used physiological recording systems is included. This technical report concludes with a summary of our present work and provides an outlook to future projects.**

***Index Terms*—wireless energy transmission, implant, biotelemetry, RFID, sensor node**

I. Introduction

The assignment of full implantable biosensors gain increasing importance in scientific and medical applications [1]. Most current recording systems are based on the use of wire connections between the neural targets and external recording devices. These physical connections either restrain the animal's movement or require recording under anesthesia, both of which affect neuronal responses [2]. In order to study neuronal signal processing in more unconstrained natural settings and to directly evaluate correlations between neuronal and psychophysical data, a wireless and implantable system for long-term analysis of complex neuronal networks is necessary.

The challenges of such fully implantable recording systems include device miniaturization combined with energy efficient integrated circuits and sufficient power supply. This can be achieved by current Wireless Resonant Energy Link (WREL) technology. This report describes the technical requirements for wireless energy transmission in fully implantable wireless sensor node systems and the design for an experimental setup.

II. Present Recording Systems

The recording of physiological signals from awake animals can occur on different ways:

- Subcutaneous recording systems
  In earlier systems recordings from awake animals were conducted subcutaneously [3] which required physical connections between the percutaneously placed recording electrodes and the actual recording system. However, this method strongly restricts the movement of the animal and causes undesired stress reactions. Moreover, long-term experiments are limited because of the risk of infection in open skin areas. To avoid damage to the hard-wired recording system, housing and behavioral training are restricted to single animals.

- Inductively powered systems
  In systems that rely on inductive power supply the recording hardware is placed under the skin and coupled to the external device via magnets. Power and commands are sent to the subcutaneous components via an inductive (coil-to-coil) wireless link [4]. This kind of energy transmission is based on magnetic influence and works only over small distances (mm to a few cm). Moreover, wired connections from the external device to the computer are used for data transmission and powering. Again, such a system restrains the animal's movements and natural behavior and affects the evaluation of neuronal correlates for psychophysical behavior.

- Battery-powered systems
  Implantable battery-powered devices [5] are first approaches to the development of fully implantable recording systems. Most of these devices have only a limited life time. Depending on the energy consumption recording durations are limited up to a few hours after which reimplantions are necessary.

- Wireless powered systems
  WREL technology allows the power supply of fully-implantable recording devices via radio frequency transmission. For the transfer of small amounts of data that has low energy requirements, radio frequency identifier (RFID) technology is used [6]. However, while regular RFID technology transmits currents in the range of only a few microamperes, new RFID based passive low-frequency interface (PaLFi) transponders transfer currents in the milliampere range.

III. WREL Technology

The WREL technology is used for wireless power transmission and relies on electromagnetic resonators. Source and receiving coil are coupled wirelessly via air. A given resonance frequency induces current on the receiving side that is stored

in a buffer capacitor. Air coupling results in a limited operating distance for power transmission. Recent research using large scale resonators reports power transmissions of up to 60 W with approximately 45% efficiency over distances in excess of two meters [7]. For the transmission of small amounts of energy in the range of microamperes, RFID can be used for the direct power supply of ultra-low-power devices. However, for the use of high efficient implantable sensor nodes currents in the range of milliamperes are required. For secure power supply the use of rechargeable batteries necessary. Modern PaLFi transponders from Texas Instruments (TI) have an integrated power management that allows to charge the battery of the implanted system. Unlike the usual battery powered systems, this modern technology provides a long term life time of the sensor node and, thus, reduces the frequency of reimplantations.

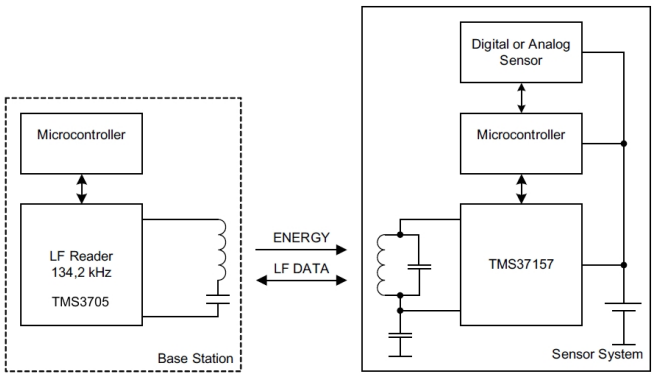


Fig. 1. PaLFi Setup, modified from [8]

IV. System Design

For wireless energy transmission we developed a system that is based on WREL technology and consists of two separate devices, the base station and the PaLFi transponder (Fig. 1). The microcontroller (MSP430F1611 8 MHz, 10 KB RAM, 48 KB ROM) from TI in combination with the SmartOS operating system [9] controls the RFID integrated circuit (TMS3705 from TI) of the base station. By driving the source coil the TMS3705 induces current in the receiving coil and allows the simultaneous transmission of both charge and data. The PaLFi (TMS37157 from TI) of the transponder transfers the energy to the rechargeable lithium coin cell (Panasonic VL-3032). The battery charge current is about 2 mA. This relatively low amount of current is sufficient to recharge the battery. Because of the limited charge rate (maximum 1/25 C or 1/25 h^{-1}), the VL-3032 can be charged with a maximum current of 4 mA. This avoids the risk of overheating and possible damage to the battery.

V. Conclusion and outlook

The power management of the TMS37157 in combination with a rechargeable lithium coin cell allows the long term use of implantable sensor nodes. Depending on the functional parameters (e.g., sample rate, data transfer rate) this technology allows two kinds of operations: In low power systems continuous recording of physiological signals can be combined with simultaneous power supply. In high fidelity systems with higher power consumption recording and recharging can be conducted in alternating cycles.

After successful benchmark testing, the described technology for wireless power transmission will be integrated into our wireless sensor node that is designed for electrophysiological recordings. This system will be evaluated in animal models for congenital and acquired deafness.

References

[1] Tan EL, Pereles BD, Horton B, Shao R, Zourob M, Ong KG. *Implantable Biosensors for Real-time Strain and Pressure Monitoring*. Sensors Basel Sensors. 2008; 8(10):6396-6406

[2] Tang X, Orchard SM, Liu X, Sanford LD. *Effect of varying recording cable weight and flexibility on activity and sleep in mice*. Sleep. 2004; 27(4):803-10

[3] Chestek CA, Gilja V, Nuyujukian P, Ryu SI, Shenoy KV, Kier RJ, Solzbacher F,Harrison RR, Shenoy KV, Stanford Univ. Stanford, CA, *HermesC: RF wireless low-power neural recording system for freely behaving primates*. IEEE Trans Neural Syst Rehabil Eng. 2008; 17(4):330-338

[4] Riistama J, Väisänen J, Heinisuo S, Harjunpää H, Arra S, Kokko K, Mäntylä M, Kaihilahti J, Heino P and Kellomäki M, et al., *Wireless and inductively powered implant for measuring electrocardiogram*. Medical and Biological Engineering and Computing. 2007; 45(12):1163-1174

[5] Lapray D, Bergeler J, Dupont E, Thews O, Luhmann HJ, *A novel miniature telemetric system for recording EEG activity in freely moving rats*. J Neurosci Methods. 2008; 168(1):119-26

[6] Yeager D, Holleman J, Prasad R, Smith JR, Otis B, *NeuralWISP: A Wirelessly-Powered Neural Interface with 1-m Range*. IEEE Transactions on Biomedical Circuits and Systems (Submitted)

[7] Kurs A, Karalis A, Moffatt R, Joannopoulos JD, Fisher P, Soljac M, *Wireless Power Transfer via Strongly Coupled Magnetic Resonances*. Science 2007; 317:83-86

[8] TEXAS INSTRUMENTS INC., Dallas (USA): *PASSIVE LOW FREQUENCY INTERFACE DEVICE WITH EEPROM AND 134.2 kHz TRANSPONDER INTERFACE*. 2009.

[9] Baunach M, Kolla R, Mühlberger C, *Introduction to a Small Modular Adept Real-Time Operating System*. In: Distributed Systems Group, 2009 editor 6. Fachgespräch Sensornetzwerke, Aachen, 2007; p.1-4

Tuontu: A Tool for Evaluating the Impact of Wireless Sensor Network Design Alternatives

Barbara Staehle, Markus Leimbach, and Dirk Staehle

University of Würzburg, Institute of Computer Science, Würzburg, Germany
{bstaehle, markus.leimbach, dstaehle}@informatik.uni-wuerzburg.de

***Abstract*—With an increasing complexity of applications running on top of a wireless sensor network (WSN), more alternatives for the WSN design arose. It is however rather difficult to asses how and at which price in terms of money or decreased quality of service, design alternatives increase the system performance. In this work we introduce a concept which is able to quickly answering likewise questions, namely the task-based resource consumption modeling. It is the heart of the framework Tuontu which allows to easily determine if an application is feasible in a given WSN deployment and which performance is to expect.**

I. Introduction

For efficiently and successfully setting up or configuring a WSN, a potpourri aspects have to be considered which are as different as hardware design choices, node deployment, protocol configurations, energy efficiency, costs, or end-user expectations. In 2004, Römer and Mattern [1] formalized this problem by describing a 14 dimensional WSN design space. Six years later, there are even more factors to consider as more sophisticated applications and hardware options appeared. Most of those factors important for setting up a WSN from scratch like e.g. sink placement [2] have been addressed by theoretical works. Many of those results are however not applicable for practical deployments, as not all used assumptions hold in reality. More helpful for configuring a productive WSN are manufacturer deployment guidelines [3] or experiences from WSN deployment campaigns [4].

A user which installs a WSN for environmental monitoring is often neither able to use an optimal combination of deployment strategies and protocols configurations, nor is she satisfied with general statements. Instead, she would simply like to know which advantages and disadvantages a certain design option has for her WSN application. Those insights need not be as detailed to require a lengthy simulation calibrated with hardware data [5], but should be more accurate the ones provides by an Excel spreadsheet [6].

In this paper, we therefore pave the way towards a new concept for evaluating WSN design alternatives by introducing the *task-based resource consumptions modeling* approach. The first pillar of this concept is to abstract the WSN to an amount of resources which are offered by the deployed nodes. The second pillar consists of decomposing each application into tasks whereof the resource consumptions are easily to determine. Those ideas are the core of a framework we call *Tuontu* as it allows to quickly analyze the impact of design alternatives and thereby helps users struggling with the question whether "To Use Or Not To Use" a certain feature.

This paper is structured as follows: In Section II we discuss related approaches. The task-based resource consumption model is introduced in Section III. Implementation details on Tuontu are given in Section IV. Section V contains numerical results illustrating the potential of our idea. We conclude and give an outlook on our next steps in Section VI.

II. Related Work

A plethora of theoretical works on WSN optimization exist which propose thoroughly evaluated algorithms. However, environmental or hardware constraints are in general not included in those studies and therefore a challenge for a practical implementation. In the paper of Bogdanov et al. [2] for example, a base station positioning algorithm which optimizes the energy efficient operation of a WSN with energy harvesting nodes is introduced. As no restrictions on the base station locations are given, this algorithm is not applicable for an outdoor WSN deployment where base stations can only be deployed at locations where a power supply and a broadband Internet access are available. A network engineer can however still alter some protocol parameters or configure the interval at which the sensor nodes collect data. Consequently she would be interested in the trade-offs involved in this decision.

The size of the design space makes it impossible to deploy and test all possible configurations. A low-level simulation framework like the one presented by Hurni and Braun [5] could be adapted to the properties of the used hardware and used for a study revealing the impact of the degrees of freedom. The two major drawbacks of a likewise approach are however that the adaptation would be difficult and very likely not feasible by an end-user and require lengthy simulation studies in order to capture all interactions. At the other edge of the spectrum are application notes or helpful hints for a successful WSN deployment. Barrenetxea et al. [4] for instance share their experience from a number of WSN deployment campaigns. This allows to avoid obvious mistakes but does not help a person willingly to build up an own sensor network to rate the trade-offs of the specific decisions. This problems is partly attacked by the manufacturer Crossbow Technology

This work was funded by the Federal Ministry of Education and Research of the Federal Republic of Germany (Förderkennzeichen 01 BK 0800, GLab). The authors alone are responsible for the content of the paper.

Inc. which provides guidelines for building a WSN based on application specific criteria [3]. The aforementioned material gives e.g. advices on the recommended number of gateways for a given number of sensor nodes. Most users are however not satisfied with such general statements, as they have no change to rate, whether an additional gateway is worth the price of a faster data delivery and increased system lifetime without actually setting up a WSN.

Answering such questions is possible with Tuontu, as task-based resource consumptions modeling strives the balance between a quick but imprecise analytical solution and an accurate but lengthy and complex simulation. Note that Tuontu is no stand-alone optimization tool for a perfect WSN configuration, but does allow to quickly assess whether the use of a more detailed simulation of the impact of a certain design factor is worth the effort or not.

III. Task-Based Resource Consumption Modeling

The two pillars of task-based resource consumption modeling are described in Section III-A and Section III-B. Firstly, a WSN is abstracted to an amount of resources. Secondly, each application running on top of the WSN into tasks whereof the resource consumptions are easily to determine. It thereby becomes possible to estimate the resource consumptions of an application and thereby to rate whether and how well it runs on a given WSN deployment. Please note that an exact analytical model is out of scope of this paper, in the following, we just use a formal language to sketch our main ideas.

A. Network Abstraction

For abstracting a WSN deployment $\mathcal{W}$, we define $\mathcal{N}$, with $N = |\mathcal{N}|$, to be the set of nodes in $\mathcal{W}$ which can be (Internet) gateways, relay or sensor nodes. The resources of $\mathcal{W}$ provided by the nodes are storage capacity, available energy and sensing capabilities. The *resource state* of $\mathcal{W}$ is hence defined as $Z =< S, E, P > \in \mathbb{R}^N \times \mathbb{R}^N \times \mathcal{P}^N$. The energy and storage resources are given by real numbers, whereas $\mathcal{P}$ denotes the set of perceptions which may be collected from the environment and depends on the sensors installed on the nodes. p_i describes how node i perceives its environment, i.e. which characteristics of the physical environment the node can capture. A gateway or relay node i does not have sensors attached. It can only report on its own operation condition, i.e. $p_i = \{\texttt{nodestate}\}$. Unless a sensor node is damaged, its perception is a non-empty subset $\mathcal{P}$. A typical sensor node perception could be $p_i = \{\texttt{nodestate}, \texttt{humidity}, \texttt{temperature}\}$.

The *physical condition* $C =< L, O > \in \mathbb{R}^{3N} \times \mathcal{O}^N$ of a WSN gives the node locations and operation state. While the column vector L is constant for networks without mobile nodes, the column vector O holding the operational state of the nodes is changing in accordance with the node activities. The possibilities for $o_i \in \mathcal{O}$ are given by the state machine used for abstracting the functionality of node i.

B. Applications and Tasks

To determine if an application α is feasible on WSN $\mathcal{W}$ and if yes, how it changes the network state Z, we analyze how much resources its execution requires. For this purpose, we decompose it to a set of tasks $\alpha = \{\tau_1, \tau_2, \ldots, \tau_n\}$, where tasks are basic functionalities which can be accomplished by $\mathcal{W}$ and whereof we can simply determine the resource requirements. An example of a simple application α_0 is to let each sensor node report one temperature reading to the nearest base station. Hence, each node has to execute the tasks τ_m of measuring a temperature, τ_s to send it to its next hop and, τ_f if necessary forward data. Each task τ is characterized by its functionality which determines which task is required for which application and by its resource requirements, $r_\tau =< e_\tau, s_\tau, p_\tau >$. The functionality of τ_m for instance, is to measure and to store the temperature. Its resource consumptions are given by $r_{\tau_m} =< e_{\tau_m}, s_{\tau_m}, \texttt{temperature} >$. The upper half of Fig. 1 depicts the principle of application decomposition.

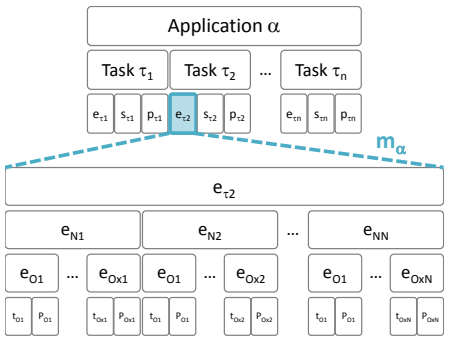


Fig. 1. Task-based energy consumption modeling

The computation of the task resource requirements r_τ and the mapping to the individual sensor nodes requires clearly more details on the task functionality and on the network configuration characterized by C. This is formalized by the application-specific mapping function m_α:

$$m_\alpha : \quad (\tau, \mathcal{W}) \quad \mapsto Z_\tau =< S_\tau, E_\tau, P_\tau > \qquad (1)$$

In the lower half of Fig. 1 is exemplarily depicted how the energy consumption, E_{τ_2} of task τ_2 are computed as the joint node energy consumptions e_{N_i}, where $1 \leq i \leq N$. e_{N_i} in turn, is the addition of the energy consumptions e_{o_j} of each of the operational states $1 \leq j \leq x_i \in \mathcal{O}$ node i is in during the task execution. Finally, the operational state energy consumptions e_{o_j} are computed as the product of state power consumptions P_{o_j} and the time t_{o_j}, node i spends in state o_j. Note that this model allows to capture different hardware equipments or modeling applications where not all nodes do the same thing, as only some nodes need to measure and or process data while others do nothing or are required for forwarding purposes only.

The effect of an application α on $\mathcal{W}$ is determined by the *application execution function*

$$E : \quad (\alpha, Z) \quad \mapsto e(\tau_n, e(\tau_{n-1}, ..., (e(\tau_1, Z)))). \qquad (2)$$

The result of E are just the resources which remain after the execution of the tasks whereof α consists. The *task execution function* accounts for the task resource consumptions

$$e : (\tau, Z) \mapsto Z' \quad = \quad Z - Z_\tau \qquad (3)$$

which is for $1 \leq i \leq N$ defined as

$$z_i - z_{\tau_i} =< e_i - e'_i, s_i - s'_i, p_i \cap p'_i >, \quad (4)$$

where the resource reductions of the individual nodes are determined by m_α. The "perception" resource does not need to be reduced, as sensors are not changed by the execution of tasks. Instead, e would yield a resource state, where one sensor node has an empty perception set. A simple sanity check if $E(\alpha, Z)$ has still non-negative and non-empty entries is hence sufficient for checking whether α is feasible on $\mathcal{W}$ or not.

IV. TUONTU

Tuontu uses the previously introduced concepts, in order to determine, if and with which performance an application may successfully be executed on a given WSN deployment. For each executed task, the resource consumptions are computed according to the previously introduced model, but we include random factors to model the imperfectness of hardware and the harshness of the environment. Tuontu is implemented in Java and intentionally kept modular to make it easy to include more design factors which describe the design of a certain WSN deployment than the ones we came up for our initial design and which we review in Section IV-A. The performance metrics which are currently implemented are described in Section IV-B.

A. Factors under Consideration

In this section we walk through the plethora of factors characterizing a WSN deployment we considered for Tuontu in a bottom-up fashion. The *deployment area* characterizes the size of the area to monitor and optionally candidate locations for the node positions. The *deployment strategy* gives the location of the nodes. As an abstraction of more or less sophisticated placement algorithms, we consider the sensor nodes to be either deployed on a regular grid, in a random fashion or clustered according to the importance of the area to monitor. Another factor influencing the physical network condition C and thereby the functions m_α and e is *the used hardware*. The resource consumption of any task strongly depend on the used hardware. As a starter, we use the Crossbow eKo node [7] as a role model to derive the node state machine and the corresponding power and time consumptions. We also adopt eKo characteristics for properties like size of the RAM and flash memory, sensing capabilities and the networking stack. Furthermore are the possibilities for the *energy supply* inspired by the eKo capabilities. We assume the most common setup where the sensor and relay nodes in the field have 2 AA batteries, whereas the gateways are mains powered. Additionally, the user could decide to augment the nodes with an energy harvesting unit or to go out and exchange batteries if necessary. The amount of energy gathered by the solar energy harvesting process is modeled to be normally distributed over the daily sunshine duration and parameterized according to [9] and [7].

The *networking stack* depends most often on the used hardware. A packet-level simulation is not our goal, we therefore use an abstract networking stack model. We use the IEEE 802.15.4 [8] physical layer channel model together with a shadow fading component parametrized to result in an average link length slightly larger than 250 m [7]. As a low-power MAC protocol, we abstract a solution similar to CSL proposed by the upcoming 802.15.4e [10]. CSL enables low-power multi-hop communication at the price of an increased delay by periodic channel scans each σ seconds which is also the length of the preamble proceeding each packet. The routing topology is abstracted to a minimum hop topology which requires a certain amount of energy to be constructed and to be self-healing. On application layer, we abstract the possible *applications* to do either periodic data reporting, to report the occurrence of random events occurs or to answer user inquiries. The *data sampling period* δ gives the length of the interval between two periodical activations of the sensors. The *application intelligence* determines to which degree the nodes do process the data. At the moment we namely consider the effects of a simple data aggregation protocol, where each forwarding node has to wait for the data packets of its children in order to forward them together with its own data.

B. Performance Metrics

The *network lifetime* is clearly the most important metric for a WSN design. As countless "lifetime" definitions exist, we use the time when 50% of all nodes have run runs out of energy as "lifetime" which is in any case an indicator for the network longetivity regardless if the WSN is still functional after this period or not. The *application layer performance* is characterized by the quality of data and the data delay. For this study, we use the average data delivery delay as application layer performance metric. It is estimated as the product of the path length and the sum of the preamble length σ and the packet transmission time and additionally includes the effect that packets might have to wait for being aggregated. The third, metric are simply the monetary *costs* of the deployment.

V. RESULTS FROM A FACTORIAL DESIGN STUDY

The goal of Tuontu is to give insights how a given WSN deployment can be optimized. For this purpose, we distinguish between *hard factors* which can not be influenced and *soft factors* which can be adapted. In this section we illustrate the potential of our idea by reporting on the results of a factorial experiment which visualizes to what degree and at what price soft factors may influence the system lifetime.

A. Experimental Setup

We assess the influence of the soft factors wherefore we show exemplary "high" and "low" values in Table I by repeating the same experiment for each of the resulting 2^5 design points in 50 different WSN topologies. One network snapshot consists of 50 sensor nodes randomly spread in a 500×500 m square with the gateway(s) at its corner(s). All factors not mentioned in Table I are considered to be "hard"

and parametrized as discussed in Section IV-A. The input files for Tuontu summarize this setup and are publicly available[1].

TABLE I
LOW (-) AND HIGH (+) SOFT FACTOR LEVELS

soft factor	level (-)	level (+)
number of gateways G	1	4
energy source	battery	solar
data aggregation	off	on
sampling interval δ	5 min	15 min
channel scan period σ	5 s	20 s

One experiment consists of running an application mix of regular data reporting, random event detection and answering user queries on top of the WSN. The experiment ends either after 2 years, or when 50% of all nodes are out of energy.

B. Numerical Results

At the end of each experiments, the metrics system lifetime, average packet delay and cost are collected. The influence of factor x on the system performance in terms of metric y is characterized by its *main effect* $e_x(y) = (\bar{y}_{x^+} - \bar{y}_{x^-})/2$, where $\bar{y}_{x^+}$ and $\bar{y}_{x^-}$ denote y averaged over all design points where x is at its high level and low level respectively. $e_x(y)$ hence simply expresses which average impact setting x from its low value to its high value has, regardless all other factors. Fig. 2 visualizes the main effects of soft factors on the system metrics. The 95%-confidence intervals demonstrate, that most of the effects are statistically significant.

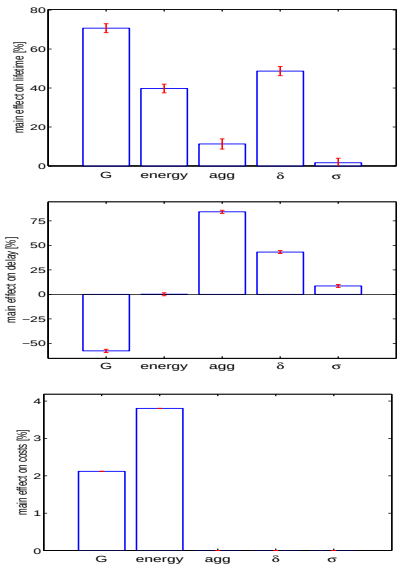


Fig. 2. Main Effects of Soft Factors

As the absolute main effects depend strongly on the WSN configuration, we show normalized main effects only in order to allow a better comparison. For actually deciding about a certain WSN configuration, the absolute values are of course necessary. This form of representation is however suitable for pointing out promising optimization direction. Fig. 2 hence confirms the intuitive assumption that choosing for all soft factors the high instead of the low value always increases the system lifetime. It also shows that some factors have a stronger influence than others and that the negative effect of some factors are different. More precisely has a larger number of gateways the strongest impact on the system lifetime, as this would lead to shorter paths, thereby reducing not only the forwarding load of the sensors , but also the average packet delivery delay. This is hence a promising way for increasing the system lifetime and performance, if the user is willingly to pay the price for the hardware. As we assumed the price for 50 solar panels to larger than the one for three additional gateways, using energy harvesting instead of batteries would in this case be the more expensive option, but this solely depends on those chosen numbers. Note however also that cost neutral options, like the use of data aggregation, longer sampling periods or also longer channel scans are also suitable for increasing the system lifetime. A closer analysis of the interactions of different parameters and implementations for those combined factors could hence be an interesting optimization option.

VI. CONCLUSION AND OUTLOOK

This work introduced the idea of task-based energy consumption modeling for wireless sensor networks. It is the heart of Tuontu, a tool for evaluating the impact of WSN design decision on the system lifetime and performance. Results from an exemplary factorial design study demonstrate the soundness and applicability of our idea. The refinement of Tuontu is ongoing and includes the integration of more factors and metrics for allowing a holistic WSN optimization and extensive parameter studies.

ACKNOWLEDGMENTS

The authors would like to thank Clemens Mühlberger and Marcel Baunach for help on technical questions and Phuoc Tran-Gia for his support and valuable comments.

REFERENCES

[1] K. Römer and F. Mattern, "The Design Space of Wireless Sensor Networks," *IEEE Wireless Communications,*, vol. 11, no. 6, 2004.
[2] A. Bogdanov *et al.*, "Power-aware Base Station Positioning for Sensor Networks," in *INFOCOM*, HongKong, China, 2004.
[3] G. Baleri, "Guidelines for WSN Design and Deployment," Crossbow Technology, Inc., 2008.
[4] G. Barrenetxea *et al.*, "The Hitchhiker's Guide to Successful Wireless Sensor Network Deployments," in *SenSys*, Raleigh, USA, 2008.
[5] P. Hurni and T. Braun, "Calibrating Wireless Sensor Network Simulation Models with Real-World Experiments," in *Networking*, Aachen, Germany, 2009.
[6] Crossbow Technology, Inc., http://www.xbow.com.
[7] Crossbow Technology, Inc., "eKo Pro Series, eKo Components - For Environmental Monitoring," Data sheet, 2009.
[8] IEEE Computer Society, "IEEE 802.15.4 Standard for Information technology," 2006.
[9] J. Taneja *et al.*, "Design, Modeling, and Capacity Planning for Micro-Solar Power Sensor Networks," in *IPSN*, St. Louis, USA, 2008.
[10] IEEE Computer Society, "IEEE P802.15.4e/D0.01 Draft Standard for Information technology," 2009.

[1] http://www3.informatik.uni-wuerzburg.de/staff/bstaehle/tuontu/

T-SIM: A Simulation Enviroment for Dynamic Wireless Sensor Networks

Christian Huisinga* and Jens Kamenik*
OFFIS Institut für Informatik
Escherweg 2
26121 Oldenburg, Germany
Email: christian.huisinga@uni-oldenburg.de
jens.kamenik@offis.de

Axel Hahn
Department of Computing Science, Carl v. Ossietzky University
Ammerländer Heerstr. 114-118
26129 Oldenburg, Germany
Email: hahn@wi-ol.de

***Abstract*—The simulation of wireless sensor networks (WSN) is a powerful possibility to evaluate WSN software and their protocols in an early design phase. This paper demonstrates T-SIM, a simulator-extension that provides different mobility models, an easy to use graphical interface and a logging system for the TinyOS 2 simulator TOSSIM. The interaction of T-SIM and TOSSIM will be shown as well as the architecture of T-SIM. An example scenario based on a mobile localization application demonstrates the abilities of the graphical user interface and the features of the logging system of T-SIM.**

***Keywords*- wireless sensor networks; TOSSIM; TinyOS; WSN simulation**

I. Introduction

The simulation of wireless sensor networks (WSN) is a powerful possibility to evaluate WSN software and their protocols in an early design phase. TinyOS [1] is a widespread operating system for WSN and provides a simulation environment called TOSSIM [2]. Via a programming interface this simulator enables the testing of TinyOS applications for defined WSN settings. If sensor nodes do not change their positions over the time, the link qualities between nodes are static and have to be defined once before starting simulation. But, todays WSN applications have to cope with mobile sensor nodes where the position of the nodes changes over the time. For example, if a forklift truck in a logistic scenario needs to communicate with its surrounding WSN [3]. To simulate such mobile applications with TOSSIM, different application specific mobility models are needed. Furthermore, when positions of transmitter or sender change, the radio link qualities in the simulation of mobile applications have to be updated frequently by a radio propagation model.

In this paper an architecture for a TOSSIM-extension called T-SIM will be shown. T-SIM (a TOSSIM Simulation Tool with Mobility Models) provides different mobility models embedded in an easy to use graphical interface for simulation control. After the discussion of the architecture of T-SIM, the ease of T-SIM will be demonstrated by an example implementation of a mobile localization application.

* Supported by the German federal state of Lower Saxony with funds of the European Regional Development Fund (ERDF) within the scope of the research project CogniLog.

II. Related Work

A common simulator is ns-2 [4] that has extensions for using it as a WSN Simulator. For ns-2 a WSN application must be converted to a ns-2 specific programming language. In contrast to ns-2, TOSSIM [2] is specifically designed for the WSN operating system TinyOS. The great advantage of TOSSIM is that the application code developed for TinyOS can be used for the TOSSIM simulator as well and therefore reduces the design effort. The following environments are visualization and simulation tools comparable to T-SIM:

TinyViz [2] was the first graphical user interface for TOSSIM. It works with the version 1 of TinyOS [1]. TinyViz does not work with the current version 2 [5] of TinyOS. Octopus [6] is a monitoring, visualization and control tool for real WSN. The information between the WSN and the visualization tool is exchanged by serial messages and needs an adapted WSN application to work. An extended version of TOSSIM (included a serial forwarder) called TOSSIM Live [7] can be used with Octopus. In contrast SimX [8] is a integrated WSN simulation and evaluation environment. SimX uses TOSSIM to simulate the soft- and hardware and includes the following features: topology manipulation per drag and drop, time control, variable watch and sensor input control. For JTossim there is no publication available at the moment but the source code from the project homepage [9]. The homepage states: "JTossim is a Graphical User Interface (GUI) to TOSSIM, the TinyOS simulator. It allows to define simulation parameters (like radio settings and network topology) and provides different visualization of the results of simulations." One of the major drawbacks of the described visualization tools is the lack of mobility models.

III. Architecture of T-SIM

T-SIM simulates WSNs, implemented for the operating system TinyOS [1] and for the hardware platform MicaZ [10]. T-SIM uses TOSSIM and forms together with the control unit the simulation core of TOSSIM (Figure 1). TOSSIM simulates the node application and the hardware components. T-SIM simulates a physical environment of the WSN by using the extended models. This **abstract physical environment** includes a tow-dimensional simulation area and defines the

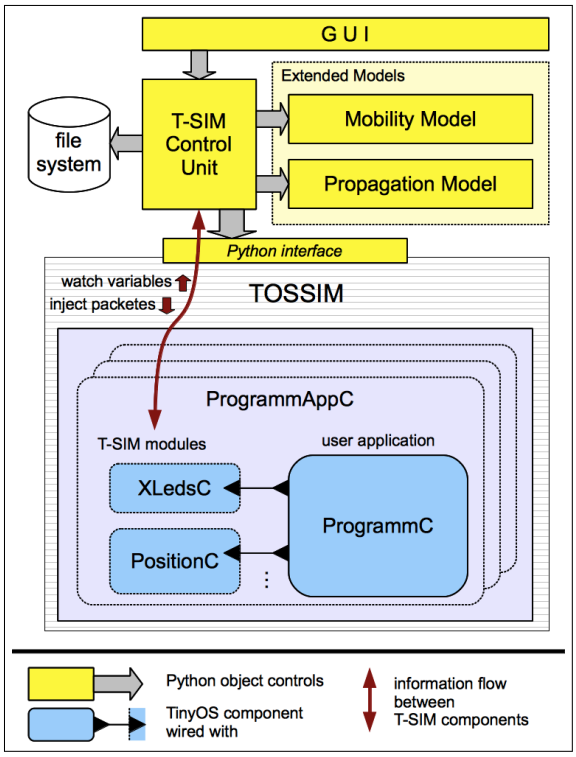


Fig. 1. Architecture of T-SIM : A graphical interface uses functions of a control unit, which controls TOSSIM and the extended models. TOSSIM simulates the TinyOS application, which uses T-SIM-Modules. The key issue in T-SIM is the information flow between the control unit and the instances of T-SIM modules. There are the injecting of packets into the simulated network and the watching of application variables.

positions of the nodes. The mobility model describes the time-dependent position of a node. The propagation model defines the link qualities between a node and all other nodes (in the following: path-loss matrix). The T-SIM control unit manages the positions of the nodes according to a chosen mobility model and calculates the path-loss matrix according to the radio propagation model of the free space. The path-loss matrix is frequently stored into TOSSIM by the control unit. The T-SIM modules provide variables necessary for the simulation. For example, the control unit watches them to display the led status in the graphical user interface (GUI). Furthermore, the T-SIM modules are able to receive injected packets. The GUI offers a life view of the WSN and eases the interaction with the WSN. The GUI, the control unit and the extended models are written in Python - the same language that TOSSIM provides an interface for. The T-SIM modules and the user application are written in nesC, the programming language of TinyOS.

A. Graphical User Interface

The GUI shows the network topology on the left side and it shows information about the running simulation and about a chosen sensor node on the right side (Figure 3). A node in the topology view has its corner marked with a circle that states the physical position. On the top of sensor node the led status is shown and on the bottom additional user information is shown. The GUI uses functions (e.g. start and stop) of the control unit to control the simulation and needs to monitor variables of the T-SIM modules to display the led status and other information. So, the WSN application has to supply variables containing this information by including T-SIM modules like XLedsC (Figure 1). Furthermore, the GUI can be used to create or modify the network topology at design time. While running the simulation the GUI allows to add or move nodes, to define the simulation area or to choose a mobility model for a node. Additionally it allows to observe application variables. All this configurations and options can be stored into a configuration file.

B. Control Unit

The main function of the control unit is built of a simulation loop that simulates the hard- and the software of sensor nodes as well as the physical environment of the WSN.

In T-SIM one pass of the simulation loop is defined as one **simulation step**. To simulate the hard- and software, the control unit uses TOSSIM and it uses the extended models to calculate the path-loss matrix. This path-loss matrix together with the positions of the nodes form the physical environment. Figure 2 shows the simulation loop as a state machine. In the state machine the variable identifiers mean:

t: the current simulation time of TOSSIM
t_0: start time of a simulation step
Δt: minimum duration of one simulation step
t_r: real duration of a simulation step

State 1 is to initialize the simulation step and to save the start time t_0 of the new simulation step. At state 2 the control unit orders TOSSIM to run simulation events for a minimum time step Δt. Because TOSSIM is an event-driven simulator the simulation time for one event is not predictable. So, after an event has been processed, T-SIM pauses the simulation of hard- and software if the minimum time step (Δt) is reached or has been exceeded. In the last case more than the simulation time Δt has passed, resulting in (1).

$$t_r = t - t_0 \tag{1}$$

After the simulation of hard- and software via TOSSIM the position of nodes and the path-loss matrix (the physical environment) have to be updated. State 3 updates the speeds of all nodes by using the chosen mobility model. State 4 calculates the positions $\vec{p}$ by (2).

$$\vec{p} = \vec{p}_0 + t_r\vec{v} \tag{2}$$

In state 5 the propagation model object calculates the radio path-loss matrix. Afterwards the control unit stores the radio path-loss matrix into TOSSIM. The duration t_r represents the time interval for updating the physical environment and for storing the log data (state 6).

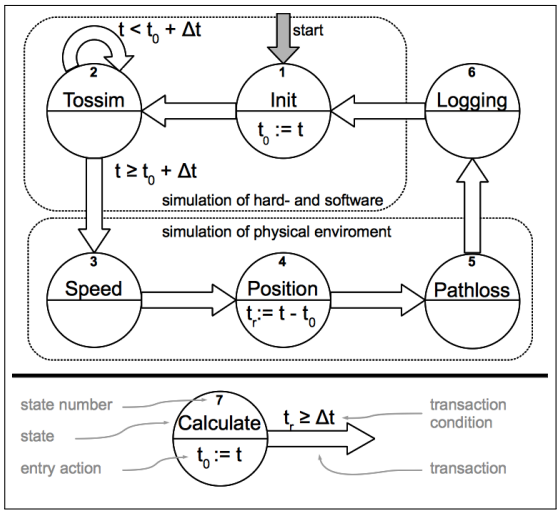


Fig. 2. Inner loop of T-SIM Control Unit. A simulation step comprises: 1) initialize the simulation step, 2) simulation of soft- and hardware, 3) calculation of speeds of all nodes, 4) calculation of positions of all nodes, 5) calculation of the radio pathloss between the nodes, 6) log data

C. T-SIM Modules

The T-SIM modules are a framework for the user application and allow communication between the control unit and the node application. In this paper two T-SIM modules are described: XLedsC and PositionC. XLedsC provides led control for the node application and is essential for displaying led status in the GUI. The T-SIM module PositionC is essential to store position information to a node via the GUI and provides the localization algorithm for the node applications. The T-SIM modules do not have to be modified for using them in a real WSN.

1) XLedsC: XLedsC is an extension of the TinyOS-module LedsC. It has an additional variable that is used for storing the led status. The GUI reads this variable to show the led status.

2) PositionC: PositionC provides the current position of the nodes. Nodes are stationary ($v_i = 0$) or mobile ($|v_i| \geq 0$). Stationary nodes know their own position. If the network simulation is started or the topology changes, the simulation core sends messages with position information to stationary nodes. PositionC receives and stores these messages. Furthermore, the mobile nodes determine their positions by analyzing the measured RSSI values on a regular basis.

The T-SIM module PositionC executes the localization by the following steps:

1) The mobile node sends a short sequence of messages.
2) The stationary node receives the messages from the mobile node and detects the RSSI values.
3) The stationary node sends the averaged RSSI value and its own position back to the mobile node.
4) The mobile node receives these messages from stationary nodes in its range and calculates its position.

To calculate the position from the RSSI values and the positions of stationary nodes PositionC uses the Weighted Centroid Localization algorithm (WCL) [11].

IV. Extended Models

The extended models in T-SIM represent the physical environment surrounding the sensor nodes. The environment includes the mobility model object and the propagation model object (Figure 1).

A. Propagation Model

The propagation model object gets the node positions and returns the radio path-loss between two nodes. It uses the free-space path-loss propagation model, according to the Friis' free space transmission equation [12]. In this equation the remaining power at the receiver depends on the distance between sender and receiver. Typically the wave length, the transmitting power and the gains are constant in a WSN. So, the received signal strength is only a function of the transmitter-receiver-separation. This propagation model is also called unit discs, because (in two dimensions) the equipotential lines of received signal strength are unit discs around a transmitter.

In embedded devices, the received signal strength is defined as ratio to a reference power (typically $1\ mW$) in a dB-scalar and it is called Receive Signal Strength Indicator (RSSI) [12].

B. Mobility Models

The mobility model object gets the current position and current speed of a node and returns the updated speed. The implemented mobility models [13] are:

- Random Walk
- Random-Waypoint
- Linear: A constant speed model.
- Stationary: No movement.

The Random Walk mobility model generates a random speed value and a random direction. The implemented Random Walk model calculates the speed of a node by (3).

$$\vec{v} = \begin{pmatrix} v_x \\ v_y \end{pmatrix} = \begin{pmatrix} v \cdot \sin\varphi \\ v \cdot \cos\varphi \end{pmatrix} \tag{3}$$

$$v = z\left(v_{min}, v_{max}\right), \quad \varphi = z\left(0, 2\pi\right) \tag{4}$$

$z\left(a, b\right)$ is a uniform distributed random value: $a \leq z < b$. The Random Waypoint mobility model generates a speed from the current position $\vec{p}$ to a random target position $\vec{p}_T$ by (5).

$$\vec{v} = \frac{\left(\vec{p}_T - \vec{p}\right)}{I} \tag{5}$$

$$\vec{p}_T = \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} z\left(0, x_{max}\right) \\ z\left(0, y_{max}\right) \end{pmatrix} \tag{6}$$

I is the constant update time of the speed of a node in the simulation.

V. Example Application

Figure 3 shows the GUI with a localization application running. Nine stationary nodes are arranged as a grid and one mobile node is moving according to the chosen linear-movement mobility model. The simulated node application uses the described T-SIM modules PositionC and XLedsC. The user information shown at the bottom of the sensor node is

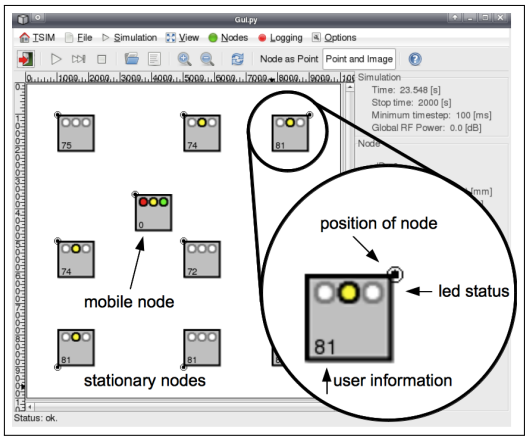


Fig. 3. The graphical user interface of T-SIM displaces the network topology and nodes' led status on the left side and network/node information on the right side.

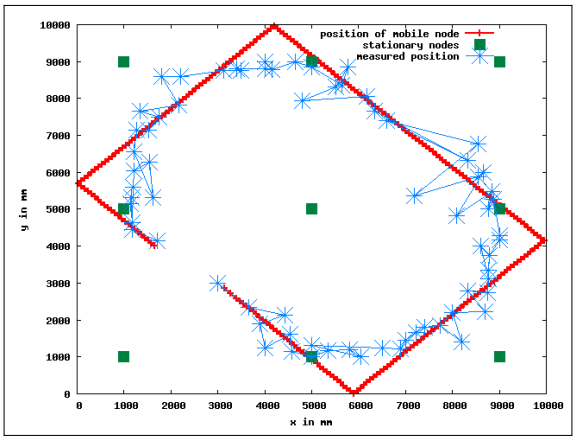


Fig. 4. Simulation results of the example localization application: the track from the mobile node vs. measured position of the mobile node

TABLE I
EXAMPLE RESULTS OF THE LOCALIZATION

	$d(\vec{p}_L, \vec{p})\,/mm$
mean deviation	745
maximal deviation	2249
mean square deviation	850

the absolute value of the radio signal strength indicator (RSSI) between a stationary node and the mobile node. The simulation area size is 10 m $\times$ 10 m and the stationary nodes are placed according to Figure 4. The logging system of T-SIM stores the positions of the mobile node, calculated in the control unit, and further application variables. In two of the further application variables the T-SIM module PositionC writes the results of the localization. Figure 4 shows the output of the simulation. One graph shows the track from the mobile node and the other graph shows the measured position derived from the RSSI data. Example results for the deviation of real vs. measured positions are shown in Table I. The distance between the stationary nodes is $\Delta p = 4\ m$ and $d(\vec{p}_1, \vec{p}_2)$ is the distance between $\vec{p}_1$ and $\vec{p}_2$. So, the ratio between the mean deviation of the position and the distance between two stationary nodes, is given by (7).

$$\frac{d(\vec{p}_L, \vec{p})}{\Delta p} \approx 0.186 \quad (18.6\ \%) \tag{7}$$

VI. CONCLUSION & FUTURE WORK

The T-SIM user is able to read the led status directly from a simulated node comparable to a real WSN node. It is an easy way to debug WSN applications and to manipulate them at runtime. To manage moving network nodes T-SIM includes mobility models and a radio propagation model that forms a (simplified) physical environment.

The radio propagation model in T-SIM is a rough estimation for the real physical environment of a WSN. For a more realistic simulation a more sophisticated radio propagation model is needed. It is also feasible to couple node positions in the simulation with the node position of a real network. To simulate an active-dynamic WSN (e.g. a node belongs to a robot) the simulator has to be extended in a way that the node application is able to actively change its own position. A further improvement would be to provide physical interactions, for example by simulation of an actuator.

REFERENCES

[1] P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer, and D. Culler, "Tinyos: An operating system for sensor networks." Springer Verlag, 2004.
[2] P. Levis, N. Lee, M. Welsh, and D. Culler, "Tossim: Accurate and scalable simulation of entire tinyos applications." New York, NY, USA: ACM, 2003.
[3] C. B. D. Ommen, J. Kamenik, and A. Hahn, "A system-architecture for robotic movements of goods – approaches towards a cognitive material flow system," *ICINCO*, 2009.
[4] ISI, "The network simulator – ns-2," [June 2009]. [Online]. Available: http://www.isi.edu/nsnam/ns/
[5] P. Levis, D. Gay, V. H, J. hinrich Hauer, B. Greenstein, M. Turon, J. Hui, K. Klues, C. Sharp, R. Szewczyk, J. Polastre, P. Buonadonna, L. Nachman, G. Tolle, D. Culler, A. Wolisz, T. U. Berlin, C. Inc, and A. R. Corpration, "T2: A second generation os for embedded sensor networks," Tech. Rep., 2005.
[6] R. Jurdak, A. Ruzzelli, A. Baribirato, and S. Boivineau, "Octopus: Monitoring, visualization, and control of sensor networks," 2009.
[7] C. S. Metcalf, "Tossim live: toward a testbed in a thread," 2007.
[8] W. S. University, "Simx: an integrated sensor network simulation and evaluation environment," [June 2009]. [Online]. Available: http://sensorweb.vancouver.wsu.edu/research/simx.html
[9] Soureforge.net, "Jtossim," [June 2009]. [Online]. Available: http://jtossim.sourceforge.net/
[10] C. Technology, "Micaz." [Online]. Available: www.xbow.com/
[11] J. Blumenthal, R. Grossmann, F. Golatowski, and D. Timmermann, "Weighted centroid localization in zigbee-based sensor networks," 2007.
[12] T. S. Rappaport, *Wireless Communications: Principles & Practice*, 1996.
[13] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," vol. 2, pp. 483–502, 2002.

An Algorithm for Fast Symmetry Reduction in Symbolic Model Checking

Christian Appold
Chair of Computer Science V
University of Würzburg
Würzburg, Germany
Email: appold@informatik.uni-wuerzburg.de

***Abstract*—Sensor networks are concurrent systems which often consist of a large number of sensor nodes and which can also possess a lot of symmetries. A technique to verify such concurrent systems is model checking. But unfortunately it suffers from the state-space explosion problem. Symmetry reduction can be used to combat this problem for concurrent systems with replicated components. A successful model checking technique is symbolic model checking by using BDDs. The combination of symmetry reduction and symbolic model checking by using BDDs suffered a long time from the prohibitively large BDD for the orbit relation. Dynamic symmetry reduction calculates representatives of equivalence classes of states dynamically and thus avoids the construction of the orbit relation. In this paper, we present an efficient model checking algorithm based on dynamic symmetry reduction. Our experiments show that the algorithm is very fast and allows the verification of larger systems.**

I. Introduction

With the growing dispersion of concurrent systems, e.g. through the use of sensor networks or multi-core CPUs, the need for reliable methods for their verification increases. A successful technique for the verification of concurrent systems is temporal logic model checking [4], [14]. Model checking is an automated formal verification technique, where properties are formulated in a temporal logic (like CTL [2] or LTL [12]). Unfortunately model checking suffers from the state-space explosion problem. This especially appears in the verification of concurrent systems, where the size of the state-space grows exponentially with the number of components. Concurrent systems often contain many replicated components and they frequently possess a lot of symmetries. Symmetry reduction techniques [8] have been developed to exploit those symmetries and to combat the state-space explosion problem. In many cases significant savings in memory and time can be achieved by using them.

Symmetry reduction techniques exploit symmetries by restricting state-space search to representatives of equivalence classes of states. One key problem of symmetry reduction which is difficult to solve is to calculate that states are in the same equivalence class. This problem is known as the orbit problem. With the help of the orbit relation, model checking can be done with a bisimilar quotient structure over the equivalence classes (see e.g. [5], [7]). In symbolic model checking with BDDs, which has been very successful in the verification of large systems, exploiting symmetry is complex. The reason therefore is that the orbit relation has to be represented as a BDD. The size of this BDD is exponential in the minimum of the number of components and the number of states per component for many frequently occurring symmetry groups [5].

A method which avoids to build the orbit relation and where orbit representatives are calculated dynamically during fixpoint iterations is dynamic symmetry reduction [6]. There transition images are computed with respect to the unreduced transition relation and successor states are immediately afterwards mapped to the corresponding orbit representatives. Dynamic symmetry reduction as presented in [6] uses a single BDD for the transition relation which contains all transitions of every component of the input program. In [3] the authors showed how a partitioned transition relation can be used instead. Therewith they have been able to verify systems which could not be verified by using a single unpartitioned transition relation, because it would be intractably large.

In this paper we propose an efficient symbolic model checking algorithm for forward reachability analysis, which uses dynamic symmetry reduction. Our algorithm does not use only a single transition relation. Instead, we always store simultaneously only the transition relation of one component of a concurrent system. Therewith our algorithm is able to verify systems where the whole transition relation cannot be build due to memory exhaustion. Through the combination of component-wise execution and full exploration of new states for one component before the execution of the next component we achieve considerable runtime improvements for dynamic symmetry reduction. Also the component-wise execution of transitions helps to implement state symmetries (see [7]) efficiently. Especially in the verification of systems with many replicated components big runtime savings can be achieved by using them. An integration of state symmetries in our efficient model checking algorithm and further experimental results can be found in [1]. For our verification experiments we used and extended the symbolic model checker Sviss [16], which implements symbolic symmetry reduction methods.

The rest of the paper is organized as follows. In the next section we give an introduction to model checking (II-A), symmetry reduction (II-B) and dynamic symmetry reduction (II-C). In Section III we present our fast model checking algorithm for dynamic symmetry reduction, before we present

experimental results which confirm the efficiency of our algorithm in Section IV. The paper closes with a conclusion and an outlook to future work.

II. Background

A. Model Checking

Model checking [4], [14] is an automatic technique to verify finite state concurrent systems. Given a finite state model describing the behavior of a system and a property, a model checker determines if the property is satisfied by the model. The finite state model of a system is usually described in the form of a *Kripke structure*.

Definition 1: Let AP be a finite set of atomic propositions. A Kripke structure M over AP is a quadruple $M = (S, R, L, S_0)$, with the following components:

- S is a nonempty, finite set of states,
- $R \subseteq S \times S$ is the transition relation,
- $L : S \rightarrow 2^{AP}$ is a function, which maps each state in S with the set of atomic propositions which are true in that state and
- $S_0 \subseteq S$ is the set of initial states.

Properties are in general specified in a temporal logic. Examples of temporal logics are CTL [2] and LTL [12].

B. Symmetry Reduction

This section gives a short introduction to symmetries in model checking. For further information see e.g. [10] or [5]. A Kripke structure is symmetric if it is invariant under certain transformations of its state-space. Permutations are used to define symmetries of a Kripke structure. Given a non-empty set X, a permutation of X is a bijection $\pi : X \rightarrow X$. We extend π to a mapping $\pi : R \rightarrow R$ on the transition level of a Kripke structure by defining $\pi((s,t)) = (\pi(s), \pi(t))$.

Definition 2: A permutation π on S is said to be a **symmetry** of a Kripke structure $M = (S, R, L, S_0)$, if:

- R is invariant under $\pi : \pi(R) = R$,
- L is invariant under $\pi : L(s) = L(\pi(s))$ for any $s \in S$, and
- S_0 is invariant under $\pi : \pi(S_0) = S_0$.

The symmetries of M form a group under function composition. A model M is said to be **symmetric**, if its symmetry group G is non-trivial. A group G of symmetries induces an equivalence relation $\equiv_G$ on the states of M by the rule $s \equiv_G t \Leftrightarrow s = \pi(t)$ for some $\pi \in G$. The equivalence class of a state $s \in S$ under $\equiv_G$, denoted $[s]_G$, is called the $orbit$ of s under the action of G. The relation $\equiv_G$ is called orbit relation. The orbits can be used to construct a $quotient$ Kripke structure M_G.

Definition 3: The quotient Kripke structure M_G of M with respect to G is a quadruple $M_G = (S_G, R_G, L_G, S_G^0)$ where:

- $S_G = \{[s]_G : s \in S\}$ (the set of orbits of S under the action of G),
- $R_G = \{([s]_G, [t]_G) : (s,t) \in R\}$ (quotient transition relation),
- $L_G([s]_G) = L(rep_G([s]_G))$ (where $rep_G([s]_G)$ is a unique representative of $[s]_G$),
- $S_G^0 = \{[s]_G : s \in S_0\}$ (the orbits of the initial states S_0 under the action of G).

The quotient structure M_G is smaller than M, if G is non-trivial. In [5] the authors show that by choosing a suitable symmetry group G, model checking can be done by using M_G instead of M, which often leads to considerable savings in memory and time.

C. Dynamic Symmetry Reduction

Dynamic symmetry reduction calculates orbit representatives dynamically during state-space traversal. Therewith the computation of the BDD for the orbit relation, which often is of intractable size, can be avoided. In dynamic symmetry reduction an abstraction operator α is used instead of the quotient transition relation, where the orbit relation is essentially embedded in. The operator α is applied to the result of image operations with the unreduced transition relation R. It is an abstraction operator which dynamically maps states that result from an image computation to their corresponding representatives. Equation 1 shows the formal definition of α. Depending on the underlying group of symmetry, the implementation of the abstraction function α has to be adapted.

$$\alpha(T) = \{rep_G([t]_G) \in S_G : \exists t \in T : (t, rep_G([t]_G)) \in \equiv_G\} \quad (1)$$

If a global state is no orbit representative, dynamic symmetry reduction swaps bits in the BDD variable order to gain the corresponding unique representative of the state. The authors of [6] say that swapping of bits in the BDD representation dominates efficiency of dynamic symmetry reduction and because of efficiency reasons they propose to use bubble sort which swaps only adjacent elements. Further information and an extension of the dynamic symmetry reduction principle to full CTL model checking can be found in [6].

III. Our fast Model Checking Algorithm

In this section we present our fast symbolic forward reachability analysis algorithm for dynamic symmetry reduction. For simplicity the algorithm is presented here for components of only one component type. An extension to multiple different component types can be found in [1]. The pseudo-code of the algorithm is shown in Listing 1. First of all, the BDD named *Init* is initialized with the initial states of the verification model. The initial states are immediately sorted (line 2) by using the abstraction function α of dynamic symmetry reduction (see section II-C). Therewith we achieve that the first forward image computation explores only successors of symmetry reduced states. Next, one BDD for successor states during forward image computation (*Successors*) and one BDD that later saves all states which have been reached during the state-space traversal (*Reached*) are generated and both initialized with the initial states (see line 3). Then one BDD for the transition relation (*TransRelation*) and another BDD

that stores states which have been reached during the current exploration of a component (*newExplored*) are generated (see line 4). The array of BDDs *toExplore* in line 6 stores for each component the states which have still to be explored for this component. The value of *numComponents* is the number of components that exist in the verification task. At the beginning, *toExplore* is for every component initialized with the sorted initial states.

In line 10 a loop starts and will be executed until there is no component that has any further states to explore. In the loop the transition relation for the currently active component is build on-the-fly (line 12). By always storing the transition relation only for the component for which forward images are calculated at the moment we are able to verify systems which cannot be verified by using a single transition relation, because a single transition relation would be too large to be build (see e.g. the peterson mutual exclusion protocol in section IV). In line 15 a loop begins which is executed as long as new states can be found for the currently active component. Inside the loop forward images ($Image_R(Successors)$) are calculated with states that have not been explored for the component so far. Afterwards unique representatives of the successor states are computed (line 17). Representatives which have not been visited during state-space traversal are saved in the BDD *Successors* and further explored for the component.

In dynamic symmetry reduction exploration of states always starts from symmetry reduced states. By execution of transitions for only one component, less changes of these symmetry reduced states can occur than by using the whole transition relation with all components for forward image computation. Therefore fewer swaps are needed to canonicalize these successor states, which reduces the time for their canonicalization. The multiple consecutive application of the forward image computation for one component has the advantage that in this way successor states often can be canonicalized considerably faster. After the full exploration of a component, all newly found states for one component are added to *toExplore* of the other components. Therewith *toExplore* can contain a large amount of states if the component which executes transitions changes. Necessary BDD swaps then can be used for a larger amount of states simultaneously. Together, as our experimental results confirm (see section IV), considerable runtime improvements can be achieved.

In line 22 and 23 the discovered new states are added to *toExplore* of the other components of the system. Whenever all components have explored their states (the loop in line 11 has finished), the algorithm tests, if there still is a component which has to explore some states. If no such component can be found, all states which are reachable from the initial states have been found and the algorithm terminates. The correctness of the algorithm follows from the fact, that every newly discovered global state is added first to *newExplored* and after the full exploration of a component to *toExplore* of all other components. Therewith forward images of this state are calculated for the component which discovered this global state and for all other components.

```
1  BDD Init = initialStates();
2  Init = α(Init);
3  BDD Successors, Reached = Init;
4  BDD TransRelation, newExplored = Empty();
5  bool finish = false;
6  BDD toExplore[numComponents];
7
8  for(i=(numComponents-1);i>=0;i--) {toExplore[i] = Init;}
9
10 while(finish == false) {
11  for(i=(numComponents-1);i>=0;i--) {
12   TransRelation = buildTransRel(i);
13   Successors = toExplore[i];
14
15   while(Successors != Empty()) {
16    Successors = Image_R(Successors);
17    Successors = α(Successors) & !newExplored & !Reached;
18    newExplored |= Successors;}
19
20   Reached |= newExplored;
21   for(z=(numComponents-1);z>=0;z--) {
22    if(z != i) {toExplore[z] |= newExplored;}
23    else {toExplore[z] = Empty();}}
24
25   newExplored = Empty();}
26
27  finish = true;
28  for(n=(numComponents-1);n>=0;n--) {
29   if(toExplore[n] != Empty()) {finish = false;}}}
```

Listing 1. Pseudo-code of our forward reachability analysis algorithm

IV. Experimental Results

For our verification experiments we have used a computer with an Intel Pentium Core 2 CPU with 2.4 GHz and 3 GB main memory by using a single core. The verification experiments have been done with the symbolic model checker Sviss, which uses the Cudd BDD package [15]. For our experiments we disabled dynamic variable reordering of BDDs. As variable order for the bits of the components in the BDDs we have chosen the variable order concatenated:

concatenated: b_{11} ... b_{1logl} b_{21} ... b_{2logl} ..... b_{n1} ... b_{nlogl}

Here b_{ij} denotes the jth bit of component i and l is the number of local states of a component. In the following table the number of components in the verification experiments can be found in the column *Problem* after the name of the verification benchmark. *Number of BDD Nodes* is the largest number of live BDD nodes that appeared during a verification experiment. This is the memory bottleneck of a verification experiment, because the model checker has to store this number of BDD nodes to finish verification successfully. Time is the runtime of a verification experiment, where s, m and h are abbreviations for seconds, minutes and hours.

The first testcase for which experimental results are shown in Table I is a simple mutual exclusion example. There every component has only the three local states *non-critical (N)*, *trying (T)* and *critical (C)*. There are state changes from *non-critical* to *trying* and from *critical* to *non-critical* which can be executed without restrictions. Also there is a global id-sensitive variable *tok*, which ranges over process indices. Its value is set nondeterministically to a process index, if a component executes a state change from *critical* to *non-critical*. Only the process whose id currently is the value of the global id-sensitive variable is allowed to make a state change from *trying* to *critical*. For our verification experiments

	Old Dynamic Symmetry Reduction		Our Fast Algorithm			Old Dynamic Symmetry Reduction		Our Fast Algorithm	
Problem	Number of BDD Nodes	Time	Number of BDD Nodes	Time	Problem	Number of BDD Nodes	Time	Number of BDD Nodes	Time
Mutex 200	337,709	4:25m	54,684	22s	CCP 10	358,127	2:49m	69,462	52s
Mutex 400	2,044,109	56:32m	189,702	3:53m	CCP 20	2,429,642	1:41h	355,394	43:37m
Mutex 600	2,932,995	4:39h	405,133	13:34m	CCP 25	4,424,644	5:35h	651,706	2:47h
Mutex 800	5,190,561	14:10h	700,120	33:35m	CCP 30	7,220,011	14:36h	1,100,968	8:36h
MCSLock 10	24,251	3s	9,333	2s	Peterson 6	81,931,144	3:22m	186,246	22s
MCSLock 20	143,715	2:43m	55,013	1:20m	Peterson 8	mem ov	-	2,385,546	10:24m
MCSLock 40	786,310	1:41h	446,849	1:04h	Peterson 10	mem ov	-	12,810,763	1:40h
MCSLock 60	2,087,657	15:56h	1,744,207	12:00h	Peterson 12	mem ov	-	66,938,967	13:22h

TABLE I
RESULTS OF OUR VERIFICATION EXPERIMENTS

we used the property that no two processes are in the state *critical* simultaneously. The verification results show that large runtime and memory improvements can be achieved by using our new model checking algorithm.

In Table I also experimental results of verification experiments with MCSLock, a modified variant of the list-based queuing algorithm from [9], can be found. For our experiments we here used the property that no two processes can possess the lock at the same time. The experimental results show that for this testcase also significant runtime improvements can be achieved by using our algorithm. Additionally Table I shows experimental results for the CCP cache coherence protocol. It refers to a cache coherence protocol developed from S. German (see for example [13]). There our algorithm is nearly twice as fast as the previous dynamic symmetry reduction algorithm. Also the memory requirements could be reduced significantly. The peterson mutual exclusion protocol [11] in Table I is a protocol, where entry to the critical section is gained by a single process via a series of $n-1$ competitions. There is at least one looser for each competition and the protocol satisfies the mutual exclusion condition, since at most one process can win the final competition. By using the old dynamic symmetry reduction algorithm verification experiments finished only for a maximum of six components because of the huge BDD of the single transition relation. It could have been build only for six components. Due to the component-wise treatment of the transition relation in our new model checking algorithm, we could verify the protocol for twelve components. Additionally we achieved large runtime and memory gains for six components.

V. CONCLUSION AND OUTLOOK

In this paper we have proposed an efficient symbolic forward reachability analysis algorithm for dynamic symmetry reduction. Through component-wise storing of the transition relation, we achieve the verification of systems where the use of a single transition relation has been intractably large before. Our experimental results confirm that the model checking algorithm is considerably faster for all testcases than the usage of dynamic symmetry reduction as presented by [6]. Additionally our algorithm reduces the memory requirements. In the future we will try to find an efficient scheme for component-wise handling of the transition relation and dynamic symmetry reduction for full CTL model checking. Also at the moment abstraction functions for dynamic symmetry reduction only exist for full symmetry and rotational symmetry. There we want to further enhance the applicability of dynamic symmetry reduction and of our new algorithm for other symmetry groups.

REFERENCES

[1] Christian Appold. Efficient symmetry reduction and the use of state symmetries for symbolic model checking. *CoRR*, abs/1006.1416, 2010.

[2] M. Ben-Ari, Z. Manna, and A. Pnueli. The temporal logic of branching time. In *POPL '81: Proceedings of the 8th ACM SIGPLAN-SIGACT symposium on Principles of programming languages*, pages 164–176. ACM, 1981.

[3] J. R. Burch, E. M. Clarke, and D. E. Long. Symbolic model checking with partitioned transition relations. In *International Conference on Very Large Scale Integration*, pages 49–58, 1991.

[4] E. M. Clarke and E. A. Emerson. Design and synthesis of synchronization skeletons using branching-time temporal logic. In *Logic of Programs, Workshop*, pages 52–71, London, UK, 1982. Springer-Verlag.

[5] E. M. Clarke, R. Enders, T. Filkorn, and S. Jha. Exploiting symmetry in temporal logic model checking. *Formal Methods in System Design*, 9(1-2):77–104, 1996.

[6] Allen Emerson and Thomas Wahl. Dynamic symmetry reduction. In *Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, 2005.

[7] E. A. Emerson and A. P. Sistla. Symmetry and model checking. *Formal Methods in System Design*, 9(1-2):105–131, August 1996.

[8] C. N. Ip and D. L. Dill. Better verification through symmetry. *Formal Methods in System Design*, 9(1-2):41–75, 1996.

[9] J. M. Mellor-Crummey and M. L. Scott. Algorithms for scalable synchronization on shared-memory multiprocessors. *ACM Transactions on Computer Systems*, 9:21–65, 1991.

[10] A. Miller, A. Donaldson, and M. Calder. Symmetry in temporal logic model checking. *ACM Computing Surveys (CSUR)*, 38(3):8, 2006.

[11] G. L. Peterson. Myths about the mutual exclusion problem. *Inf. Process. Lett.*, 12(3):115–116, 1981.

[12] A. Pnueli. A temporal logic of concurrent programs. *Theoretical Computer Science*, 13:45–60, 1981.

[13] A. Pnueli, S. Ruah, and L. Zuck. Automatic deductive verification with invisible invariants. pages 82–97. Springer, 2001.

[14] J.-P. Queille and J. Sifakis. Specification and verification of concurrent systems in cesar. In *Proceedings of the 5th Colloquium on International Symposium on Programming*, pages 337–351, London, UK, 1982. Springer-Verlag.

[15] F. Somenzi. *CUDD: CU Decision Diagram Package, Release 2.4.2*. University of Colorado at Boulder, http://vlsi.colorado.edu/ fabio/CUDD/.

[16] T. Wahl, N. Blanc, and A. Emerson. Sviss: Symbolic verification of symmetric systems. In *Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, 2008.

Mobility Assisted Positioning in Wireless Sensor Networks

[Extended Abstract]

Hannes Frey and Martin Schwier
University of Paderborn
hannes.frey@upb.de

I. Introduction

Finding the physical location of sensor nodes has already been considered in many directions. Typically, physical characteristics of radio and/or ultrasonic wave propagation are exploited to measure absolute or relative distances or orientation of nodes in Euclidean space. With that information at hand plain trigonometric computations, geometric approximations, multilateration techniques, or multidimensional scaling are applied in order to estimate the sensor nodes' positions. Refer to [1] for an overview of sensor node positioning techniques.

We consider one new such positioning technique, first described in [2], which exploits the mobility in robot supported wireless sensor networks. In the original publication of that technique a quite astounding precision has been reported. The studies of that work were based on simulation. Thus, it remains open what happens if the system is actually applied in a real world setting.

In this work we give some answers on that question. Extending the simplified method described in the original publication, we propose different techniques to determine the point when a mobile robot is closest to a sensor; which is a necessary requirement for that positioning scheme. We show that opposed to the simulation results reported, less precision is to be expected in a real world setting.

II. Positioning Concept

We assume a mobile beacon node whose current trajectory is given by a straight line. The mobile beacon knows its current position and its trajectory. It periodically sends out a message containing that information.

Consider the mobile beacon moving through a field of sensor nodes. Assume for now that the beacon is moving along one straight line. Due to path loss, for each sensor node the received signal strength is first increasing as long the beacon is moving closer to the sensor. Once the beacon passed by, the signal strength of the beacon will again decay.

Each sensor node determines the signal strength of the incoming beacon messages and uses this information to estimate the mobile beacon node's position when it was closest. It then considers itself located somewhere on the line orthogonal to the beacons trajectory which is passing through that estimated beacon position.

To eventually determine its position, a sensor has to determine the orthogonal lines from at least two non collinear beacon movements. We can either consider a single beacon node consecutively moving through the sensor field along different trajectories. We may as well assume a set of beacon nodes moving along different trajectories at the same time. In case of perfect trajectory and position information at the beacon nodes and in case the sensor node could perfectly determine the position the beacon nodes were closest, all orthogonal lines will intersect at one point; which is the sensor node's position.

However, in practice neither of that information is perfect. In particular, due to shadowing and fading the signal strength at the receiver does not necessarily reflect the sender receiver separation. Thus, finding the position the beacon node was closest by just measuring the beacon's signal strength is not an obvious task.

In this work we propose techniques for smoothing the curve of noisy signal strength measurements over time. Moreover, having the curve of signal strength measurements over time – consisting either of the raw data or the smoothed curve – we may have more than one point in time where that curve is maximum. We describe several ways who to finally decide from that curve when the beacon node was closest. Finally, we describe a simple scheme to estimate a sensor nodes position from several independent closest beacon position estimates.

A. Smoothing Noisy Signal Strength Measurements

1) Curve Smoothing: One method we consider is the *simple moving average (SMA)*. A data point is given by the statistical mean value over the last n measurements. The parameter n determines the smoothness of the resulting curve. For $n = 1$ the curve just reflects the measured points. The larger n the smoother the curve and the more likely it will be to get a curve with one single maximum data point. However, a large n also means that the curve peak will be shifted to the right, i.e., the estimated point when the beacon is closest to the sensor nodes will be delayed.

The delayed peak point can be compensated by using the *second order simple moving average (SMA2)*. Let x_i be the data points we get from the simple moving average method. We apply the SMA method on those data points again using the same parameter n as we used for computing the x_i values. With that we obtain the averaged values y_i. The point estimates z_i of the SMA2 method are then given by $z_i = 2 \cdot x_{i-1} - y_{i-2}$.

The SMA and SMA2 methods require memorization of the n previous signal strength measurements. If the scheme is applied on sensor nodes with limited available memory other methods with less memory demands are of interest as

well. As an alternative to SMA and SMA2 we also consider the *exponential weighted moving average (EWMA)* where the current mean value is determined by the current measurement weighted by α plus the previous mean value weighted by $1-\alpha$. The parameter α is selected between 0 and 1.

The smaller α the smoother will be the curve over the mean values. However, as with the SMA method, for small α values the peak of the measured signal strength values will be delayed. The same correction technique we used with SMA2 can be applied to compute a *second order exponential weighted moving average (EWMA2)*.

2) Curve Fitting: According to the path loss model the signal strength at the receiver node is given by c/d^{α}, where c is a system dependent constant, d the sender receiver separation and α the environment dependent path loss exponent. With that model, given the distance h between the sensor node and the orthogonal projection of the sensor on the straight line trajectory of the beacon node, for each beacon position the signal strength at the receiver can be computed. Since we are only interested in the actual point in time the beacon was closest to the sensor node we can assume for simplicity the beacon node trajectory being the x axis and the sensor node being located somewhere on the y axis of a 2D coordinate system. Under that assumption, given the distance h and the current beacon position x on the x axis we can compute the signal strength as $c/\sqrt{x^2+h^2}^{\alpha}$. Of course the distance h and the position x are not known to the sensor. However, we can use the previous formula and try to find α, x and h such that the curve expressing the signal strength received at the sensor best fits the actual sequence of measured data. For doing that several curve fitting approaches are known. In this work we apply the Levenberg-Marquardt algorithm. Once we have found the best fit α, x and h of the predicted signal strength with respect to the measured data, the curve given by that best fit describes a smoothed version of the signal strength data which has exactly one maximum.

B. Finding the Point of Closest Distance

With exception of curve fitting, for all other methods (including just using the plain data) the curve not necessarily will have a single maximum. We consider three different methods to finally decide when the beacon node was closest. The simplest one is the *select first maximum (SFM)* method which just chooses the point in time when the considered sequence of data points is for the first time maximal.

A further variant we consider is the *median over n best (MDNB)* method where we consider all measurements which are located in the interval between maximum value and maximum value minus n. For that subset of data points we consider the times the beacon has sent that beacons. We select the median over these times to be the point in time when the distance between beacon and sensor was the smallest one. The position the mobile beacon was closest will be the position of the beacon at that time.

In addition, we consider also the *mid point over n best (MPNB)* method which is an extension of the previously described one. Using the same data set as determined by the MDNB method we find for each value in the data set the first and the last sample with that value. Then we compute the mid point of both data samples' time stamps. Finally, we compute the median over all such computed mid points and select this as the time the beacon was closest to the sensor node.

C. Determining the Position from Several Measurements

Having determined two non-collinear orthogonal lines that the sensor node is located on, the sensor node's position can then be determined by computing the intersection of both lines. If more than two orthogonal lines have been determined, a common intersection point may not be available due to inaccuracies of the methods used for finding the orthogonal lines. However, for each possible pair a unique intersection can be computed. In our positioning approach we consider all possible intersection points of pairs of the orthogonal lines. Then we compute the median over both x and y value of all intersection points. We consider the result of this computation as the sensor node position.

III. Evaluation

A. Experimental Setup

We considered an indoor experimental setting for evaluating the quality of the methods described. The experiment was conducted in our about $60m^2$ sized computer lab. A beacon node was moved along a straight line track and the receiver node was placed at different locations at or close to the orthogonal line passing the middle of that track. We used two Tmote Sky nodes, one for the mobile beacon and one for the sensor node whose location is to determined. Such nodes are equipped with a 2.4GHz CC2420 transceiver. Beacon massages were transmitted as broadcast without link layer had acknowledges. The MAC layer was plain CSMA.

We implemented a TinyOS program which periodically emits one beacon about every 10 milliseconds on the beacon node and which listens for incoming beacon messages on the other sensor node. Beacon transmission and reception is based on the ActiveMessage module. The signal strength at the receiver was measured using a CC2420 feature which allow reading out the *received signal strength indicator (RSSI)*. It provides information about the current signal strength at the antenna notwithstanding if it is due to reception of a message at that node or due to other unrelated transmissions around.

We adjusted the signal of the beacon node to 3 which is about $-25dBm$. We selected this value after some initial trials where we found that this value gives the best results in our experimental setting. It is not too weak such that it will be received at all considered beacon positions. As well it is not too strong such that there is a visible variation in signal strength when changing the mobile beacon position.

We implemented further software and hardware parts to automate the measurement process. The beacon node was attached on a toy train running at about 50 cm per second on a 6 meters straight line railway track. The antenna of the beacon node (which is soldered at the opposite site of the USB connector of the Tmote Sky nodes) pointed to the driving direction.

The train's departure and arrival time between the rail track end points were automatically measured by reed contacts at

the end of that track. The reed contacts were attached to the expansion connector of the Tmote Sky sensor node. They triggered an interrupt on the node which was then reported via serial active message to an attached PC. On the PC we used the reported interrupts which also included a time stamp of the sensor node to compute the train positions for each beacon transmitted. To further automate the measurement process, the running direction of the train was controlled by an air gap switch circuitry which let the train repeatedly move back and forth.

B. Results

First we were interested in the signal strength at the receiver node and how it may change if the mobile beacon moves along the same track for several times. We placed the node at a distance of 20 cm of the middle of the railway track. The receiving node's antenna was oriented in the same direction as the beacon node's one. Then the train with the beacon node on it passed by, driving back and forth for 37 times. Fig. 1a shows the average over RSSI values at the receiver node depicted over the train positions. The averages are given with 95% confidence intervals. What can be seen from the small size of the confidence intervals is that for each possible train position the signal strength at the receiver behaves almost deterministically. From the curve progression over train positions however it is visible that the RSSI value at the receiver is far from ideal path loss behavior. There is a general trend of increasing signal strength when the beacon node is moving toward the sensor node values from -300 to 0 on the x axis) and then decreasing signal strength when the beacon node is moving away again (values from 0 to 300 on the x axis), of course. However, due to antenna characteristics and multipath reception, RSSI can be subject to deep fades. With our selected beacon intervals and driving speed neighboring beacons' RSSI values are highly uncorrelated. In particular, at the point where the orthogonal line on the train is crossing the position of the beacon receiver, the signal strength breaks done. The measurement shows the need for methods to find the peak point of the beacon's signal strength.

We were also interested in the influence of the antenna orientation and the distance to the beacon track. Using a 25 cm distance to the beacon node's track, we considered an antenna orientation of $0°$, $120°$ and $240°$ relative to the beacon antenna. Further, using a $0°$ relative orientation of the receiver's antenna, we considered a distance of 25 cm, 85 cm, and 125 cm to the beacon track. Measurement results are given in Fig. 1b and 1c. In those figures we are just looking at a single train pass ignoring confidence intervals since we know from the previous experiment that there is almost perfect correlation between several train passes. However, from both figures we can also see that correlation of RSSI values disappears as soon the receiving node's position is changed or its antenna is rotated.

What can be seen from Fig. 1b is that the peak point will be shifted depending on the antenna orientation. Compared to the $0°$ values, the $120°$ values are increasing a little earlier when the mobile beacon gets closer. The same way the $240°$ values appear to increase a bit more delayed then the beacon passes by. Thus, antenna orientation introduces positioning errors since it may result in the peak point determined too early or determined too late.

From Fig. 1c we can see that precision of the measurement is highly influenced by the actual distance between receiver and the beacon node's track. While a clear peak is visible for 25 cm and still can be guessed for 85 cm, there is almost no peak pattern visible for 125 cm.

Now we want to elaborate the actual positioning precision which can be obtained from the investigated variants. For the following measurement we consider sensor nodes located at three close by locations for 20 cm, 45 cm, and 65 cm distances from the beacon node's moving track. Thus, in total we have nine locations which we consider. For each location we consider three possible antenna orientations of $0°$, $120°$, and $240°$. With that we model the mobile beacon passing by with different trajectory angles which eases the experimental procedure significantly. Otherwise, we would have to rotate the toy railway track around the receiving sensor node which was also not possible due to room dimensions and furniture in that room. In the following, we will just speak of the mobile beacon's driving direction bearing in mind that we model this by just turning the receiver instead of turning the beacon track around the receiver.

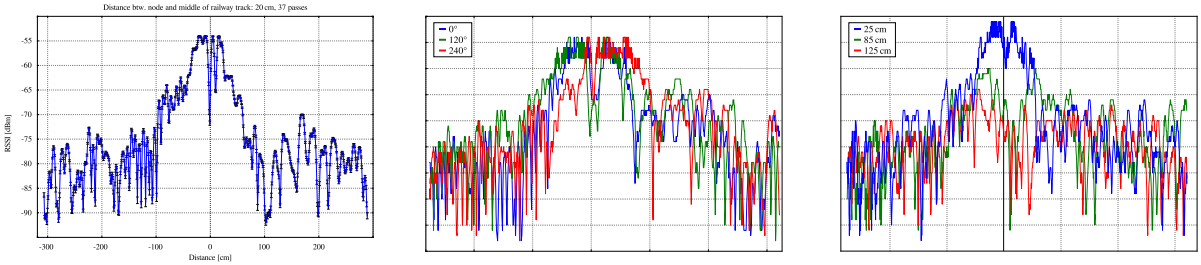
Ignoring parallel driving directions, for each of the considered nine sensor node positions we get 12 possible combinations of pairs of beacon node driving directions. For each pair we estimate the sensor node's position using the intersection of the orthogonal lines we get from the two driving directions. Thus, in total we have 108 position estimates for nine different positions. To estimate the localization precision, we compute the mean square error of the distance between the true sensor node position and the estimated one.

Fig. 2a shows that mean square error for the different variants we described in this work. For SMA and SMA2 we considered parameter $n = 50$ and for EWMA and EWMA2 we considered parameter $\alpha = 0.01$. For the MDNB and MPNB methods for finding the point of closest distance we considered parameters $n = 0$ and $n = 5$.

What can be seen form these experiments is that smoothing the curve does not improve the precision. In contrast, as can be seen for EWMA it might even degrade performance. Here, the best results were obtained when using the plain data or when using curve fitting. When using plain data, the MPNB method with parameter $n = 5$ was performing best, which also compared to all other variants was the best performing in our experiments.

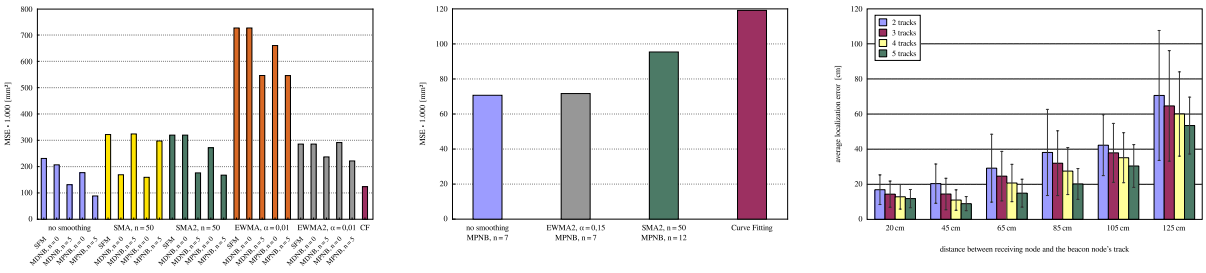
To see if the right choice of n or α can further improve the methods for finding the point of closest distance, we performed an exhaustive search and found further improvements. The best performing and their respective parameter settings are depicted in Fig. 2b. Still the MPNB method applied on the plain data directly appears to be the best choice.

To get an idea about closeness of the estimated position and the sensor's true position we considered the *average localization error (ALE)* used as well in [2]. While we could have used other metrics here as well, we choosed ALE to have a direct comparison with the results from [2]. For n position estimates



(a) Similarity of the same trips. (b) Influence of the node orientation. (c) Influence of the node distance.

Fig. 1: Signal strength at the receiving node.



(a) Errors in finding the closest beacon position. (b) Good parameter settings. (c) Average localization errors.

Fig. 2: Errors encountered for the presented methods.

the ALE it is defined as $\frac{1}{n}\sum_{i=1}^{n}\sqrt{|x_i^e - x_i| + |y_i^e - y_i|}$, where (x_i, y_i) are the true positions and (x_i^e, y_i^e) are the estimated ones.

Fig. 2c shows the average localization error including the standard deviation. We considered node distances as in the previous experiments. In addition, we also considered the larger distances 85 cm, 105 cm, and 125 cm. The results are shown for the best performing mechanism we figured out in the previous experiments. In addition to that, we also depict the localization error depending on the number of different beacon tracks. As applied in the previous experiments, for two track it means just computing the intersection of two vectors. For more than two, we get in general more than one intersection point. We then apply the method described in the previous section to estimate the sensor node position.

What can be seen from Fig. 2c is that adding more different beacon tracks can be used to improve precision. Moreover, increasing the distance between the receiving node and the beacon node's track significantly degrades the precision of the determined position.

IV. Conclusion

In this work we considered a mobility assisted positioning method, extended it with methods to cope with shadowing and fading in real environments, and evaluated the approach in a real experimental setting. We figured out that the method of the original publication – just selecting the point where the value is maximal for the first time – is way too simplified in a realistic shadowing and fading environment. We guess the reason why the authors of that publication reported quite promising simulation results (ALE of $1m$ for a distance between sensors and beacon track of up to 150 m) is due to the fact that in the original publication the approach was evaluated in a cookie cutter simulation study. Though we already proposed ways to cope with shadowing and fading in this work, such astounding precision is not supported by our experiments.

We don't think that the approach in general is not working. However, a deeper study of the method is required to finally obtain more precise position estimates. One interesting approach would be beacon strobes walking through different strengths and frequencies to better compensate shadowing and fading. Also the environment where the positioning method is applicable could further be studied. Here, we considered an indoor scenario which typically suffers from multipath propagation. It is sort of worst case scenario for positioning. When running the systems outdoors, better experimental results can be expected. We also performed such measurements and from preliminary results useful information was extracted from RSSI measurements up to 25 m distance to the beacon nodes track. However, while here the useful distance significantly improved we also observed so far that the localization error is scaled about the same compared to the indoor measurements we performed here.

References

[1] I. Stojmenovic, *Handbook of Sensor Networks: Algorithms and Architectures*. John Wiley & Sons, 2005.

[2] G. Yu, F. Yu, and L. Feng, "A three dimensional localization algorithm using a mobile anchor node under wireless channel," in *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, 2008, pp. 477–483.

Exploiting Semantic Quorum-Based Data Replication in Wireless Sensor Networks

Kinga Kiss Iakab*†, Felix Jonathan Oppermann†, and Oliver Theel
Department of Computer Science
Carl von Ossietzky University of Oldenburg
26111 Oldenburg, Germany
Email: {kinga.kiss-iakab,oppermann,theel}@informatik.uni-oldenburg.de

Jens Kamenik‡
OFFIS
Escherweg 2
26121 Oldenburg, Germany
Email: jens.kamenik@offis.de

***Abstract*—Most current wireless sensor networks (WSN) are connected to a base station that provides persistent data storage. In application scenarios without a constantly connected base station, in-network storage of the data is necessary. Usually, individual sensor nodes of WSNs are unreliable. To ensure dependable persistent storage within the WSN, redundancy of sensor data is required. Generally, data replication is used for increasing availability of data in distributed systems. The downside of data replication is increased energy consumption, which reduces the lifetime of WSNs. In this paper, we present the adaption of a semantic data replication strategy for WSNs. The approach guarantees increased availability of data in WSNs while reducing the communication costs by the exploitation of semantic properties of the application scenario. A probabilistic version of the approach is considered for further reducing the energy consumption. The example of a distributed FIFO queue is used to illustrate the approach.**

***Keywords*-wireless sensor networks; distributed data storage; semantic data replication; probabilistic data replication;**

I. Introduction

As of today, advantages in microelectronics permit to equip individual sensors with limited computing capabilities and radio interfaces. This allows the formation of large ad hoc networks to co-operatively monitor large areas or distributed phenomena. Such a wireless sensor network (WSN) is usually cheaper and more robust than the traditional approach having a small number of powerful sensors. It is envisioned to apply WSNs in a wide range of different scenarios, including application areas like habitat and structure monitoring, catastrophe management, and home automation. Sensor nodes generally have limited power supplies. Thus, the energy consumption must be reduced to extend the overall lifetime of the network as much as possible. The unique properties of WSNs demand for specific solutions to many problems and often disallow the use of traditional approaches.

Most current WSN applications relay their data to a base station outside of the network that allows a dependable persistent storage of the gathered data. Still, this approach is not suitable for all application scenarios. It is envisioned to also deploy WSNs in remote locations, for example, in habitat monitoring, where no infrastructure is available. In these scenarios the gathered data needs to be temporarily stored in the WSN itself. As individual sensor nodes are usually unreliable, it is not sufficient to store the data only at the node that generated it. To ensure dependable persistent storage in the presence of faults, it is necessary to redundantly store the data. The problem of fault-tolerant distributed data storage is already well studied for wired systems. Data replication subsumes the management of data redundancy in distributed systems, a well-known technique for increasing dependability and availability of the data. In this paper, we model a distributed system as a set of connected nodes, each managing a copy of a data object called replica. A data replication strategy provides specific operations on the replicated data object. Strict pessimistic data replication strategies ensure that no operation execution results in an inconsistent replicated data object according to a strict consistency notion like sequential consistency [1]. These operations can exploit the concept of quorums. Quorums are subsets of the set of nodes such that every two quorums corresponding to conflicting operations intersect. This ensures the mutual exclusion of these operations. Semantic data replication strategies exploit semantic properties of the data object, the operations, or the application scenario for more efficient data replication. A subclass of semantic data replication strategies concentrates on the type of the replicated data object, which can be thought of as an abstract data type, like a FIFO queue, a counter, or a dictionary.

The main contribution of the paper consists of the adaption of an existing semantic quorum-based replication strategy for WSNs. In addition, we propose an extension to exploit semantics of the application scenario and we sketch a probabilistic version of the approach. This allows to further reduce the energy consumption and thus better meet the stringent requirements of WSNs.

The remainder of this paper is organized as follows: In the next section, we review related work on different approaches for distributed data storage in WSNs employing data replication. Section III presents a novel approach for applying a semantic quorum-based replication strategy to the field of WSNs. In Section IV, the approach is illustrated by a WSN-specific example. Section V gives a brief outlook on associated

* Also affiliated with the Transilvania University of Braşov, Romania.
† Supported by the German Research Foundation (DFG), grant GRK 1076/3 (TrustSoft).
‡ Supported by the German federal state of Lower Saxony with funds of the European Regional Development Fund (ERDF), research project CogniLog.

challenges and future work. Finally, Section VI concludes the paper.

II. Related work

Data replication strategies have extensively been studied in the scope of classic distributed systems. Up to now, WSN research did not put strong emphasis on data replication.

In [2], [3], and [4], a simple Read One Write All (ROWA) [5] strategy is used to prevent query hot spots in a WSN using in-network data storage based on geographic hash tables. In [6] data replication is misleadingly formulated as the file allocation problem (FAP) [7]. The FAP optimizes the allocation of files in a distributed file system with random access to the data in terms of communication costs and storage space. Special WSN semantics are not considered. In [8] ring-based queries are used to improve energy consumption by decreasing the number of replicas depending on the spatial scope of a query. In [9] data replication is used to allow several connected WSNs to share data. By the collaboration of nodes from different WSNs each with simple tasks, a more complex shared objective can be achieved. Growth Code [10] is a new distributed data encoding technique designed to increase the persistence of sensed data. In one of the rare quorum-based approaches for WSNs [11], service directories are replicated within the elements of quorums.

Most data replication strategies used in WSNs are similar to the ROWA strategy and do not exploit the specific application scenario's semantics. On the contrary, we take a more holistic view on replication by adopting the well-studied approach of quorum-based data replication strategies adjusted to meet the specific requirements of WSNs.

III. Adapted General Quorum-Consensus

Quorum-based data replication strategies are more flexible than classic replication strategies like ROWA, and in general they can model several particular replication strategies. They allow to adjust the trade-off between operation availabilities, costs, and data consistency. In addition, quorum-based replication strategies are resilient to network partitions. Data replication implies data redundancy, which can be costly. Since energy consumption in WSNs is an important issue, data replication in this context is more challenging than in wired distributed systems. This motivates the idea of exploiting semantic properties of the replicated data object, its access operations, or the application scenario's semantics for more efficient data replication.

Herlihy introduced the General Quorum-Consensus strategy [12] for exploiting semantic properties of abstract data types (ADT) such as sets, queues, or directories back in the eighties. His general approach is suited for a system consisting of N nodes, where each node is allowed to initiate the execution of ADT-specific operations. The replicated data object of node i is represented by a partial history h_i consisting of the executed operations. An entry $(op, t) \in h_i$ is composed of the operation op (potentially with parameters) on the replicated data object and a corresponding system-wide totally ordered time stamp t. Every operation op has an associated initial quorum IQ_{op} and final quorum FQ_{op}. The role of initial quorums is to collect partial histories from nodes in IQ_{op}. Complementarily, final quorums are used to keep the nodes from FQ_{op} up-to-date.

According to Herlihy, an ADT-specific operation op is executed according to the following algorithm:

1 IQ_{op} := acquireAndLockIQ(op)
2 $H := \bigcup_{i \in IQ_{op}} h_i$
3 $H' := H \cup \{(op, t)\}$
4 FQ_{op} := acquireAndLockFQ(op)
5 foreach $j \in FQ_{op}$ do
6 $\quad h_j := h_j \cup H'$
7 unlock(IQ_{op})
8 unlock(FQ_{op})
9 return the result of op

The node initiating the execution of the operation op first searches for an available initial quorum. After an initial quorum IQ_{op} is found, the nodes from this quorum are locked. The next step is to collect the partial histories h_i of the nodes i in the quorum. Then, the union H of these partial histories is built. A new history H' is constructed that adds to the previous union history H the new entry corresponding to the executed operation. Then, a final quorum FQ_{op} for op is acquired and locked. After this, the partial histories of the nodes j from FQ_{op} are atomically updated by unifying them with H'. Finally, all nodes from the initial and final quorums are unlocked and the operation's result is returned to the caller.

Most of these steps imply communication between nodes according to a protocol like the following: 1) request messages are sent to nodes, 2) these nodes receive and process the request, and 3) they reply with a message. Depending on the quorum sizes this can lead to a large number of messages.

If two consecutive operations $op1$ and $op2$ are conflicting operations (i.e., their concurrent execution can violate the underlying consistency notion) then the following condition must hold to ensure their mutual exclusion: $FQ_{op1} \cap IQ_{op2} \neq \emptyset$.

After considering semantic data replication in the context of the General Quorum-Consensus strategy, we could further exploit other semantic information like application scenario specific properties. For example, if only one node is allowed to initiate the execution of a single instance of a particular ADT-specific operation at a time, then less consistency conflicts must be handled. In this case, no concurrent executions of this operation are possible. Therefore, communication costs can be further reduced. This makes sense in WSNs because different application scenarios have a manifold of properties which could be additionally exploited for enhancing the replication strategy.

If applying data-type-specific and application-specific replication does not satisfy the energy consumption demands, another step to improve the replication would be to use probabilistic quorums. Probabilistic quorums relax the intersection property of strict ones. On one hand, this implies smaller quorum sizes, which translates to reduced communication

TABLE I
OPERATION CONFLICTS FOR FIFO QUEUE

→	enq	deq
enq		×
deq		(×)

costs. On the other hand, the relaxation comes at the cost of relaxing data consistency. Hence, if probabilistic data consistency guarantees are sufficient then the energy consumption of our approach can be further reduced by using probabilistic quorums. For constructing consistency-driven probabilistic quorums the approach from [13] is suited.

IV. EXAMPLE: DISTRIBUTED FIFO QUEUE

In this section, we illustrate our approach with the help of a specific example from the area of WSNs. Let us consider a WSN that is deployed for a security application to detect intruders in a certain geographical area. The WSN sporadically generates events with a global scope if an intruder is detected. We assume that we cannot guarantee a constant connection between the base station and the WSN. Consequently, we need to (temporarily) store the sensed data within the network. The data storage within the network should preserve the order of the event occurrences.

These application-specific requirements are satisfied by a global FIFO queue. The General Quorum-Consensus strategy allows to implement such a global ADT in an efficient way by exploiting semantic properties of the FIFO queue operations. Typically, a FIFO queue defines two operations: the enqueue operation `enq(i:item)` that adds a new item `i` at end of the queue and the dequeue operation `deq():item` that removes the first item and returns it to the caller. As mentioned in the previous section, the replicated data object is represented by a partial history of log entries, composed of an operation (`enq(i:item)` or `deq():item`) and the corresponding time stamp. Conflicting operation sequences are `enq` → `deq` and `deq` → `deq`, because in both cases the result of the second operation depends on the action of the first one. These conflicts are marked with "×" in Table I. For ensuring the exclusion of these conflicts, the following two conditions have to be satisfied when determining initial and final quorums for the two operations: 1) $FQ_{enq} \cap IQ_{deq} \neq \emptyset$ and 2) $FQ_{deq} \cap IQ_{deq} \neq \emptyset$.

As an extension of Herlihy's original approach, we can also exploit the application-specific assumption that only one base station exists. This renders the conflict between two dequeue operations impossible. Only the base station can dequeue items (one at a time) so that we can guarantee that no concurrent dequeue operations will occur. This removes from consideration the `deq` → `deq` conflict, marked in parentheses in Table I. Hence, for the FIFO queue example, only one condition has to be satisfied, namely $FQ_{enq} \cap IQ_{deq} \neq \emptyset$. In this case, the initial quorums of the enqueue operation and the final quorums of the dequeue operation may be empty. In conclusion, communication costs can be further reduced by exploiting additional application-specific properties.

Let us consider a WSN consisting of $N = 4$ nodes with the IDs 1, 2, 3, and 4 and one base station. According to the previous considerations, there is no need for non-empty initial quorums for the `enq`-operation and non-empty final quorums for the `deq`-operation. Still, the `enq` → `deq` conflict has to be excluded. For example, the quorum sizes $|FQ_{enq}| = 2$ and $|IQ_{deq}| = 3$ can be considered, since every possible FQ_{enq} intersects with every possible IQ_{deq}. For instance, $\{1, 2\}$, $\{3, 4\}$, and $\{2, 3\}$ can be selected as final quorums for the `enq`-operation and $\{1, 3, 4\}$ as initial quorum for the `deq`-operation. In the context of the intrusion detection scenario, geographical positions of the form $<x, y>$ at which some activity was sensed are queued by the nodes and dequeued by the base station. The example from Table II illustrates the adapted approach for our WSN scenario. First, three `enq`-operations are executed using the final quorums $\{1, 2\}$, $\{3, 4\}$, and $\{2, 3\}$, respectively in this order. The union history H remains empty, because there are no initial quorums for these operations. The new history H' always contains the new log entry which is added to the partial histories of the nodes in the final quorums. Then, the `deq`-operation builds the union history H of the partial histories of the nodes from the initial quorum $\{1, 3, 4\}$. After constructing H, it can be identified which data tuple must be removed from the FIFO queue. This fact is marked by adding the corresponding log entry to the new history H'. The `deq`-operation has no final quorums, so that the execution of the operation is not added to the partial histories of some nodes. Still, the consistency of the FIFO queue is preserved with the help of the base station's history.

The addition of the base station's history is an extension of the approach to this application-specific WSN and does not exist in the original approach of Herlihy. The role of this additional history is to store all entries which were sent by the nodes when executing the `deq`-operation and also to save the `deq`-entries. The latter are not distributed to the partial histories. The nodes do not need to know about `deq`-entries, since the base station is the only one who dequeues.

For further reducing the communication costs, one can relax the one intersection condition which is left in our WSN scenario. We assume that the base station when dequeuing does not return data tuples which have smaller time stamps than the last data tuple which was returned. If the final quorums of the `enq`-operation and the initial quorums of the `deq`-operation do not necessarily intersect then some data tuples may be ignored by the base station. The degree of this inconsistency depends on the construction and selection of quorums, the length of FIFO queue and the frequencies of executing `enq`- and `deq`-operations. Our hypothesis is, that with the adequate selection of these parameters, almost no consistency loss is associated.

V. FUTURE WORK

Currently, we are working on an implementation of semantic data replication on top of TinyOS. We intend to evaluate this

TABLE II
FIFO QUEUE IN WSN EXECUTING ENQ AND DEQ

t	Operation	Node 1	Node 2	Node 3	Node 4	Base Station	H	H'
1	enq(<4,2>)	(enq(<4,2>),1)	(enq(<4,2>),1)	∅	∅	∅	∅	(enq(<4,2>),1)
3	enq(<2,3>)	(enq(<4,2>),1)	(enq(<4,2>),1)	(enq(<2,3>),3)	(enq(<2,3>),3)	∅	∅	(enq(<2,3>),3)
6	enq(<3,1>)	(enq(<4,2>),1)	(enq(<4,2>),1) (enq(<3,1>),6)	(enq(<2,3>),3) (enq(<3,1>),6)	(enq(<2,3>),3)	∅	∅	(enq(<3,1>),6)
7	deq()=<4,2>	(enq(<4,2>),1)	(enq(<4,2>),1) (enq(<3,1>),6)	(enq(<2,3>),3) (enq(<3,1>),6)	(enq(<2,3>),3)	(enq(<4,2>),1) (enq(<2,3>),3) (enq(<3,1>),6) (deq()=<4,2>,7)	(enq(<4,2>),1) (enq(<2,3>),3) (enq(<3,1>),6)	(enq(<4,2>),1) (enq(<2,3>),3) (enq(<3,1>),6) (deq()=<4,2>,7)

implementation in a prototypical outdoor deployment with 120 sensor nodes. As hardware platform we use the MAXFOR MTM-CM5000-MSP, a clone of the well-known TelosB mote.

In order to allow a functional implementation on real WSN hardware, some aspects of the approach need to be further adjusted to meet the constraints of WSN hardware. Sensor nodes usually only have very limited memory resources. The main memory is especially restricted. For example, the TelosB sensor node has a main memory size of 10 kB. In addition, TinyOS supports only static memory allocations. This limitation is challenging when the nodes have to collect partial histories from other nodes and build the union history to execute an operation on the replicated data object. This demands an implementation of the partial histories with bounded and low memory requirements. In the FIFO queue example, this is less problematic, as the `enq`-operation can be implemented with empty initial quorums and thus without building a union history. The `deq`-operation is only executed on the base station, where memory constraints are less strict.

In a real WSN, communication between two nodes may require to relay messages via multiple hops. Furthermore, messages in WSNs are expensive compared to classical distributed systems. To keep communication costs acceptable, we need to reduce the number and size of messages. A way to reduce the number of messages is to use a clever construction and selection of the quorums keeping in mind the properties of WSNs. These aspects also play an important role in the development of a consistency-driven instance for the probabilistic approach. In addition to designing a suitable quorum construction and selection, we intend to deploy caching and further compaction of the partial histories to further reduce communication costs.

The prototype will allow us to evaluate the behavior of the approach under realistic conditions. We intend to demonstrate its applicability for typical WSN scenarios and to compare it with naïve approaches, like ROWA.

VI. Conclusions

In this paper, we presented a novel approach for applying semantic quorum-based replication to the field of WSNs. The General Quorum-Consensus strategy proposed by Herlihy allows to exploit the semantics of ADT operations. As this permits to reduce the required number of messages – and thus preserve energy – this strategy seems well suited for WSNs. An adapted General Quorum-Consensus strategy was illustrated by applying it to a particular WSN scenario. In the future, we intend to further refine this approach to better meet the hardware constraints of WSNs. In addition, we are currently working on an implementation on real hardware to evaluate the approach's suitability for daily use. We hope to report on this in due time.

References

[1] L. Lamport, "How to make a multiprocessor computer that correctly executes multiprocess programs," *IEEE Transactions on Computers*, vol. 28, no. 9, pp. 690–691, 1979.

[2] S. Ratnasamy, B. Karp, L. Yin, F. Yu, D. Estrin, R. Govindan, and S. Shenker, "GHT: A geographic hash table for data-centric storage in sensornets," in *Proc. of the 1st ACM International Workshop on Wireless Sensor Networks and Applications (WSNA)*. ACM, 2002.

[3] S. Ratnasamy, B. Karp, S. Shenker, D. Estrin, R. Govindan, L. Yin, and F. Yu, "Data-centric storage in sensornets with GHT, a geographic hash table," *Mobile Networks and Applications*, vol. 8, no. 4, pp. 427–442, 2003.

[4] S. Shenker, S. Ratnasamy, B. Karp, R. Govindan, and D. Estrin, "Data-centric storage in sensornets," *SIGCOMM Computer Communication Review*, vol. 33, no. 1, pp. 137–142, 2003.

[5] P. A. Bernstein, V. Hadzilacos, and N. Goodman, *Concurrency control and recovery in database systems*. Addison-Wesley Longman Publishing, 1987.

[6] J. van Greunen and J. Rabaey, "Content management and replication in the SNSP: A distributed service-based OS for sensor networks," *Proc. of the 5th IEEE Consumer Communications and Networking Conference (CCNC)*, pp. 655–659, 2008.

[7] L. W. Dowdy and D. V. Foster, "Comparative models of the file assignment problem," in *ACM Computing Surveys 14 (2)*. ACM, 1982, pp. 287–313.

[8] B. Krishnamachari and J. Ahn, "Optimizing data replication for expanding ring-based queries in wireless sensor networks," in *Proc. of the 4th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, 2006, pp. 1–10.

[9] D. Gračanin, K. P. Adams, and M. Eltoweissy, "Data replication in collaborative sensor network systems," in *Proc. of the 25th IEEE International Performance, Computing, and Communications Conference (IPCCC)*, 2006, pp. 389–396.

[10] A. Kamra, V. Misra, J. Feldman, and D. Rubenstein, "Growth codes: Maximizing sensor network data persistence," in *Proc. of the ACM SIGCOMM Conference 2006*, 2006, pp. 255–266.

[11] V. Raychoudhury, "Efficient and fault tolerant service discovery in MANET using quorum-based selective replication," in *Proc. of the 7th Annual IEEE International Conference on Pervasive Computing and Communications (PerCom)*, 2009.

[12] M. Herlihy, "A quorum-consensus replication method for abstract data types," *ACM Transaction on Computer Systems*, vol. 4, no. 1, pp. 32–53, 1986.

[13] K. Kiss Iakab, C. Storm, and O. Theel, "Consistency-driven probabilistic quorum system construction for improving operation availability," in *Proc. of the 11th International Conference on Distributed Computing and Networking (ICDCN)*. Springer, 2010, pp. 446–458.

Exploring the Applicability of Participatory Sensing in Emergency Scenarios

(Extended Abstract)

Diego Costantini, Andreas Reinhardt, Parag S. Mogre, Ralf Steinmetz
Multimedia Communications Lab, Technische Universität Darmstadt
Rundeturmstr. 10, 64283 Darmstadt, Germany
Email: *{dcosta, areinhardt, pmogre, rst}@kom.tu-darmstadt.de*

***Abstract*—In emergency scenarios, rescuers need specific information about the affected areas. Such information are normally collected by sensors, but sometimes they could not be available because, as a consequence of a disaster, the pre-deployed infrastructure might be damaged, and rescue teams could not easily deploy new sensors in unfavorable conditions. In such situations, it is possible to exploit the victim's devices, already located within the affected area, and powerful enough to provide the needed sensing information (environmental, like smoke and temperature, and personal, like users' biometrics). Furthermore, this paradigm (so called Participatory Sensing), by providing a highly pervasive computing, can drastically save deploying costs for extra sensors. This paper presents the main research challenges expected for participatory and opportunistic sensing in emergency scenarios, and gives insights into possible solutions to tackle them.**

I. Introduction

The ongoing embedding of smartphones with sensors has led to some interesting service, like BikeNet [1], Ear-phone [2], and more. These services are based on the idea that users generate content through their mobile handsets. Such novel means of data acquisition are termed Participatory Sensing [3] or People-centric Sensing [4], and allow many yet unknown services to be realized. In contrast to traditional sensor networks, system architecture in participatory sensing has no control over users' mobility and actions. In fact, the assumptions on scalability, mobility, availability, etc., are in this case completely different [4]. Such issues have recently been faced by researchers, but in this paper we focus on application of participatory sensing to emergency scenarios, an interesting application domain which has not yet received significant attention. In this domain, requirements and constraints are more restrictive than in classic participatory sensing. We believe that using participatory sensing/actuators for emergency scenarios will enable the extension of the existing emergency response infrastructure (e.g., public warning) and also permit faster detection of emergencies with better granularity of sensing in certain scenarios. Additionally, the use of user equipment for sensing brings with it certain benefits:

- Sensors (users' handheld mobile devices) are usually carried by the users without any overhead, and hence provide an ideal platform to gather vital information about the users (who are the primary concern in emergency scenarios) and, at the same time, provide a platform to inform the users about actions to be taken or avoided in an emergency scenario.
- Deployment and maintenance costs for the sensor network are reduced drastically and are borne by the users (willingly, and without much overhead) to a great extent.
- The range of the network can potentially be extended with each new mobile handset being purchased and brought into service by the users, and given the recent numbers forecasting 970 million of smartphone users by the end of 2013 against the 100 million of 2009 [5], the amount of sensors in such a system is expected to grow very rapidly, giving the potential to sense at a granularity (both temporal and spatial) which is not possible using traditional sensor networks.

However, to realize the full potential of the above, certain important research challenges need to be addressed. In Section II we highlight some basic difference between participatory sensing in normal and emergency scenarios and we provide a concrete reference scenario to better understand the implications of our assumptions. Our contributions in Section III are: (1) an outline of the most critical and interesting research challenges detected for the reference emergency scenario, and (2) some insight into possible solutions to tackle them; Section IV discusses the related work in participatory sensing and in emergency scenarios, and finally Section V concludes the paper.

II. Participatory Sensing in Emergency Scenarios

Assumptions and constraints in an emergency scenario can be very different to traditional participatory sensing. We are going to quickly show a few examples of such differences, and to propose a concrete class of events (earthquakes, explosions, gas leaks) to give the reader a specific reference situation for the research challenges that will be described.

A. Major Differences with Traditional Participatory Sensing

The most prominent problems usually associated to participatory sensing are scalability, privacy, and battery consumption [6], [4], [7]. Scalability (in terms of amount of data and network traffic) is a critical issue due to the huge number of devices participating, because applications can potentially receive data from participating devices all over the world; privacy defines the amount of information provided by the

users, and balancing such amount to provide a useful set of information while preserving the privacy of the users is no easy task; battery consumption is an even bigger problem, because users can directly notice fast battery exhaustion and most likely block any sensing application. In emergency scenarios, though, scalability normally has a reduced impact due to the locality characteristic of the scenario. However, it cannot be completely ignored, because there could be emergency situations where a high number of people (and their devices) are gathered (i.e., during an event in a stadium). Also privacy can be seen from a different point of view, since the scope of the application making use of it is limited to a specific set of information, and the goal of such application is the immediate benefit of the participating users. Battery consumption, instead, is much more critical in emergency scenarios. What normally is just an annoyance for the users, which have to recharge the devices more frequently (often it happens daily), becomes now of paramount importance, because the battery should last as long as possible, without the chance to recharge it, until the rescuers can help the victim.

B. Reference Scenario

When a disaster situation occurs, many people suffer injuries and/or tend to panic, sometimes also hindering the work of rescue teams. These people, and especially their technological devices, can however provide much help and information about their surroundings. For example, following an earthquake or an explosion, rescue teams will have to explore and search for victims in buildings that are on fire or collapsed. Some of the victims might be unable to communicate their position, but the environment can provide useful data through sensors to locate them or describe the status of particular areas. In particular, envisioning the continuous growth of handheld devices market, we expect everybody to have a sensor-equipped device able to help the rescue teams with sensed information (such as camera feed, microphone, smoke, temperature, ...) or alarms (i.e., loud noise and blinking light to help detect the device and, hopefully, the trapped or injured owners). These devices will be an addition to the sensors already present in the environment, which may not be working anymore due to the disaster. These are the conditions we assume while addressing the following research challenges.

III. Research Challenges

To provide a system able to take advantage of personal devices within the scenario just defined, multiple research challenges must be addressed, many of them covering different research areas. We will present here those we consider most interesting and critical (see Fig. 1).

A. Devices Participation

The first important issue to solve regards the devices participation. Several problems must be faced to make sure that the users, and thus, their devices, are properly participating. Motivation and incentives for users is quite a big issue for traditional participatory sensing, because users are normally not voluntarily willing to share and spend their resources for the benefit of the community. In this case, though, being a tool to potentially save their lives, should be a good enough incentive. However, users often do not foresee a disaster, and they judge how the system impacts them every day. That is why the sensing application, to be accepted by the users, has to take into consideration the following problems, regardless of any technical reason.

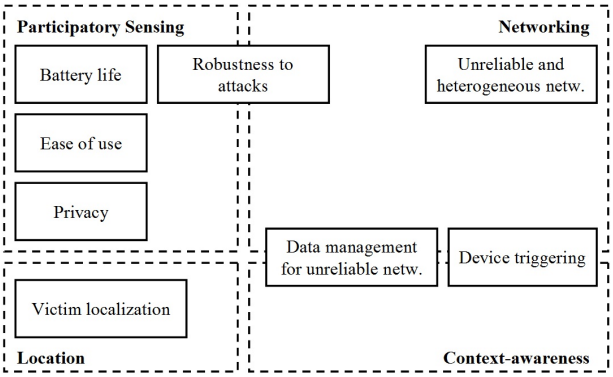


Figure 1: Logical View of the Research Challenges

1) Battery Life: Battery life is already a concern for smartphone users [8], and adding more (visible) consumption will not make any sensing application welcome on their devices. Therefore, it must be consumption-aware. But besides users' attitude, saving battery is a requirement also for the reference scenario. In fact, if we consider an earthquake or an explosion, some building could collapse, and victims might be trapped under the debris and extracted after a long time, making paramount to have the devices able to participate as long as possible. Hence, highly efficient and robust algorithms and protocols are needed.

2) Ease of Use: Further, special emphasis is needed on ease of use. Users should not note the overhead of the sensing application, as this is not the primary aim of the users neither of the handheld devices. Therefore, the application should be seamless and lightweight. Ideally, it could be seen as a safety feature offered by the device manufacturer, on which users have no power (or even awareness) at all, as it currently happens with the emergency call system of mobile phones. Giving rights on the application to the users could bring, especially with a full featured set of sensors, to misconfigurations or rejections, which go against the purpose of the sensing application itself.

3) Privacy: Since users provide sensitive sensor data, it should be clear what is disclosed, when, and to whom [9]. For the purpose of an emergency application, only the information relevant to the current emergency should be transmitted, for the duration of the emergency, and only to other devices actively or passively participating to the system.

4) Robustness to Attacks: The system should be robust to attacks of all kinds: from compromised user devices providing bogus or misleading data, to unnecessary emergency response costs due to false/malicious alarms, or misdirection of people

in real emergency scenarios. For this purpose, the system should rely on multiple sources of data and treat them before acknowledging an alarm, in particular, detecting and filtering outliers, and giving trusted devices (i.e., belonging to some trusted authority) a heavier weight for decisions.

5) Device Triggering: One main mechanism related to the previous problem is the activation of the devices. Three options are envisioned in case of disaster: (1) users manually turn their devices into emergency mode if they are able to, (2) authorities remotely trigger them, or (3) devices are able to recognize a disaster pattern and to automatically turn into emergency mode (e.g., measuring high temperature and a sudden acceleration could represent an explosion). The manual trigger is trivial, but the other two present research challenges. For example, the remote triggering requires a secure authentication mechanism to avoid malicious actions. It also requires that the triggering signal is limited to the area of the emergency, to avoid undesired propagation. The automatic emergency inference, instead, relies only on the device sensors, and extensive tests to study events patterns are required to be able to reliably infer a particular event.

B. Unreliable and Heterogeneous Networking

Once the devices are properly participating within the system, the focus must be shifted to the networking problems in hostile post-disaster conditions [10], [11]. In fact, any pre-deployed network infrastructure is not guaranteed to be fully connected at all times because of malfunctions due to the disaster, and much more mobility than in traditional networks is expected (if not by the victims V, the rescue team R is supposed to move within the area of interest - see Fig. 2), thus they can be treated as Delay Tolerant Networks (DTNs).

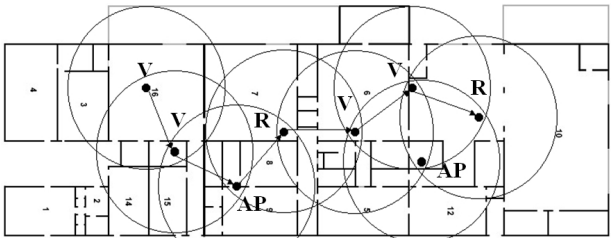


Figure 2: Heterogeneous static/mobile scenario.
R: rescuers - **V**: victims - **AP**: access points

Because of these reasons, routing solutions in DTNs over multiple wireless technologies should be investigated to cope with such assumptions. For example, bridging routing through different wireless technologies (i.e., WiFi and Bluetooth) can help solving partitioning problems, while peer-to-peer protocols can be used to address scalability issues, although scalability is expected to be a much smaller concern with respect to traditional participatory sensing, as already explained in Section II.

C. Data Management for Unreliable Networks

Assuming the unreliability of the network, data management mechanisms must be able to correctly map, spatially and temporally, sensor information collected during disconnections. They must also provide data replication and distribution. This way, nodes churn and mobility will have a smaller impact on sensor data availability and correctness. Peer-to-peer protocols could be good candidates for such tasks, and, if carefully designed, they can also mitigate the problem of scalability in crowded areas. Another important mechanism to consider in order to reduce scalability issues, is aggregation/mediation [12], [13]. Nodes can reduce the amount of data transmitted and stored (trade-off between processing power, battery consumption, and memory requirements), but they must preserve an appropriate level of informative content, thus taking into consideration Quality of Information (QoI [14]).

D. Victim Localization

Finally, victims should be located with good accuracy. Although there are countless location mechanisms proposed in literature, roughly divided between fingerprinting and triangulation/trilateration methods, they mostly have requirements/assumptions that cannot be always taken for granted. The former method assigns a position based on a set of parameters (called fingerprint), i.e., the visible WiFi access points or GSM cells, and requires the system to be trained with an initial set of fingerprints. Unfortunately, they result to be unusable when the infrastructure changes due to a disaster. The latter, instead, measures angles/distances from known points to determine the position, but again, in case of infrastructure changes and failures, it might not be possible to do it. Some examples are [15], [16], [17] for fingerprinting and [18], [19] for triangulation/trilateration. In a disaster scenario, the worst case should be assumed, and the location mechanism should work in such adverse conditions. Basically, assuming that the networking infrastructure could be absent, no area map available, and no external help (i.e., GPS or special purpose antennas) can reach the disaster area, the only "beacons" available are the victims' and rescue team's devices. Of course, every additional technological help should be exploited to improve the accuracy and reliability of the designed mechanism(s). They should try to provide an exact or relative position of the devices within the disaster area, and possibly build a map in real time as well. Locations and nodes could also be tagged with metadata representing useful information (e.g., temperature and picture taken by the target device) for the rescuers.

IV. Related Work

In literature, a number of applications relying on participatory sensing can be found. For example, NoiseTube [20] and Ear-Phone [2] use mobile phones as noise sensors, sharing geo-tagged noise pollution levels measured by the users; the BikeNet [1] application measures cyclists' movements (speed, distance, position, ...) and physical values (heart rate, galvanic skin response, or other values measurable through body sensors), and stores/shares them with other participant users; Nericell [21] uses smartphones' sensors to monitor road and traffic conditions, and reports them to a server

for aggregation. Other examples are available, but, to the authors' knowledge, none of them applied participatory and opportunistic sensing paradigm to emergency scenarios. For these scenarios, researchers normally rely on pre-deployed equipment or devices deployed after-the-fact by rescue teams, like, for example, Dilmaghani in [22], which plans to deploy wireless mesh nodes within the disaster area. A couple of works considering users' devices located within emergency areas are: SHIELD [23], which focuses on alarm propagation to trusted nearby entities, mostly in relation to localized crimes, but does not offer any automatic help to rescue teams; and WIPER [24], which provides a crisis detection mechanism by monitoring cellphones call data, searches for anomaly patterns, and proposes responses to emergency situations. Our work is different from previous ones in literature because it plans to exploit the participatory paradigm in emergency scenarios by making use of casual victims' devices happening to be in the area of the emergency even when they are not able to interact with their devices, i.e., if they are unconscious. Furthermore, it will not limit itself to the initial alarm propagation, but also to help rescue teams during the critical phases of first response and later retrieval of victims.

V. Conclusion and Future Directions

In this paper, we presented the most critical research challenges expected while applying participatory and opportunistic sensing to emergency scenarios. Having different constraints and assumptions in such scenarios, the same challenges result to be different than in traditional participatory sensing. Tackling such challenges would provide the chance to cover with sensors those areas subject to disasters which rescue teams cannot reach because pre-deployed infrastructures broke and it is not possible to deploy new sensors. In the future, we are going to work on the aforementioned open issues, starting from the devices participation and moving to networking. An additional interesting topic to investigate in relation to emergency scenarios is body sensor networks, which, especially during critical post-disaster conditions, could provide to the authorities very important data regarding victim's health.

References

[1] S. B. Eisenman, E. Miluzzo, N. D. Lane, R. A. Peterson, G.-S. Ahn, and A. T. Campbell, "The BikeNet mobile sensing system for cyclist experience mapping," in *SenSys '07: 5th international conference on Embedded networked sensor systems*. New York, NY, USA: ACM, 2007, pp. 87–101.

[2] R. K. Rana, C. T. Chou, S. S. Kanhere, N. Bulusu, and W. Hu, "Ear-phone: an end-to-end participatory urban noise mapping system," in *IPSN '10: 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*. New York, NY, USA: ACM, 2010, pp. 105–116.

[3] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M. B. Srivastava, "Participatory Sensing," in *Workshop on World-Sensor-Web: Mobile Device Centric Sensor Networks and Applications*, 2006, pp. 117–134.

[4] A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, R. A. Peterson, H. Lu, X. Zheng, M. Musolesi, K. Fodor, and G.-S. Ahn, "The Rise of People-Centric Sensing," *IEEE Internet Computing*, vol. 12, no. 4, pp. 12–21, 2008.

[5] research2guidance, "Smartphone Application Market To Reach US$15.65 Billion In 2013," Online: http://www.research2guidance.com/?p=66, 2010.

[6] L. K. Alazzawi, A. M. Elkateeb, A. Ramesh, and W. Aljuhar, "Scalability Analysis for Wireless Sensor Networks Routing Protocols," *Advanced Information Networking and Applications Workshops, International Conference on*, vol. 0, pp. 139–144, 2008.

[7] A. Ruzzelli, R. Jurdak, and G. O'Hare, "Managing mobile-based participatory sensing communities," in *Participatory Research Workshop, SENSYS 2007*, 2007.

[8] Y. Wang, J. Lin, M. Annavaram, Q. A. Jacobson, J. Hong, B. Krishnamachari, and N. Sadeh, "A framework of energy efficient mobile sensing for automatic user state recognition," in *MobiSys '09: 7th international conference on Mobile systems, applications, and services*. New York, NY, USA: ACM, 2009, pp. 179–192.

[9] K. L. Huang, S. S. Kanhere, and W. Hu, "Preserving privacy in participatory sensing systems," *Computer Communications*, vol. 33, no. 11, pp. 1266 – 1280, 2010. [Online]. Available: http://www.sciencedirect.com/science/article/B6TYP-4X4GHVG-1/2/a373cfa606785ef54d1c0b84bc0d4829

[10] B. Manoj and A. H. Baker, "Communication challenges in emergency response," *Commun. ACM*, vol. 50, no. 3, pp. 51–53, 2007.

[11] R. Dilmaghani and R. Rao, "On Designing Communication Networks for Emergency Situations," in *Technology and Society, 2006. ISTAS 2006. IEEE International Symposium on*, 8-10 2006, pp. 1 –8.

[12] T. Pham, E. J. Kim, and M. Moh, "On Data Aggregation Quality and Energy Efficiency of Wireless Sensor Network Protocols - Extended Summary," in *BROADNETS '04: First International Conference on Broadband Networks*. Washington, DC, USA: IEEE Computer Society, 2004, pp. 730–732.

[13] B. Krishnamachari, D. Estrin, and S. B. Wicker, "The Impact of Data Aggregation in Wireless Sensor Networks," in *ICDCSW '02: 22nd International Conference on Distributed Computing Systems*. Washington, DC, USA: IEEE Computer Society, 2002, pp. 575–578.

[14] P. Banerjee, "Measuring the quality of information in clustering protocols for sensor networks," in *WICON '07: 3rd international conference on Wireless internet*. ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007, pp. 1–5.

[15] F. Zaid, D. Costantini, P. Mogre, A. Reinhardt, J. Schmitt, and R. Steinmetz, "WBroximity: Mobile Participatory Sensing for WLAN- and Bluetooth-based Positioning," in *SenseApp '10: Fifth IEEE International Workshop on Practical Issues in Building Sensor Network Applications (to appear)*, 2010.

[16] B. Lakmali and D. Dias, "Database Correlation for GSM Location in Outdoor & Indoor Environments," in *4th International Conference on Information and Automation for Sustainability*, 2008, pp. 42–47.

[17] J. Kwon, B. Dundar, and P. Varaiya, "Hybrid Algorithm for Indoor Positioning using Wireless LAN," in *60th IEEE Vehicular Technology Conference*, vol. 7, 2004, pp. 4625–4629.

[18] A. Lamarca, Y. Chawathe, S. Consolvo, J. Hightower, I. Smith, J. Scott, T. Sohn, J. Howard, J. Hughes, F. Potter, J. Tabert, P. Powledge, G. Borriello, and B. Schilit, "Place Lab: Device Positioning using Radio Beacons in the Wild," in *Third International Conference on Pervasive Computing*, 2005, pp. 116–133.

[19] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The Cricket Location-Support System," in *6th Annual International Conference in Mobile Computing and Networking*, 2000, pp. 32–43.

[20] N. Maisonneuve, M. Stevens, M. E. Niessen, and L. Steels, "NoiseTube: Measuring and mapping noise pollution with mobile phones," in *ITEE*. Springer, 2009, pp. 215–228.

[21] P. Mohan, V. N. Padmanabhan, and R. Ramjee, "Nericell: rich monitoring of road and traffic conditions using mobile smartphones," in *SenSys '08: 6th ACM conference on Embedded network sensor systems*. New York, NY, USA: ACM, 2008, pp. 323–336.

[22] R. B. Dilmaghani and R. R. Rao, "A Wireless Mesh Infrastructure Deployment with Application for Emergency Scenarios," in *5th International ISCRAM Conference*, 2008.

[23] G. S. Thakur, M. Sharma, and A. Helmy, "SHIELD: Social sensing and Help In Emergency using mobiLe Devices," *Computing Research Repository (CoRR)*, vol. abs/1004.4356, 2010.

[24] T. Schoenharl, S. Member, R. Bravo, and G. Madey, "WIPER: Leveraging the cell phone network for emergency response," *International Journal of Intelligent Control and Systems*, vol. 11, p. 2006, 2007.

A Report on System and Radio Transmission Issues in a Star-Topology WSN

Sebastian A. Bachmaier
Universität Stuttgart
Stuttgart, Germany
Email: sebastian.bachmaier@iwb.uni-stuttgart.de

***Abstract*—Sophisticated methods for transmitting data in wireless sensor networks are described in literature. These include multi-hop-algorithms for radio transmission. For real-world applications, a star topology is often adequate; however, simplifying matters with respect to power consumption and software complexity.**

This report will present a star topology WSN with a flexible architecture, showing the possibilities for use in deployments. The focus of the report is a survey on system reliability issues, dealing with general failures of the real-world system, as well as with radio transmission issues. Presented general failures are meant to give the practitioner an idea of what kind of mishaps are likely to occur in a system deployment. Resulting from the radio transmission survey, the applications and limits in applications of star topology networks is appreciated.

I. Introduction

The wireless sensor network (WSN) measurement system is developed by TTI GmbH - TGU Smartmote, a spin-off of the University of Stuttgart, together with the Institute of Construction Materials and the Material Testing Institute of the University of Stuttgart. The in-house development of a WSN started several years ago with the objective of creating a customizable WSN system, suitable and designed especially for the needs of construction monitoring. After having worked with commercially available WSN nodes in a former project [1], for flexibility and cost reasons, a new platform was developed [2][3]. Sensor adaptation boards for a multitude of sensors and physical quantities had been designed, so after having laid out these sensor boards for the new platform, all the former sensors can be attached as well. In addition to the redesign of the hardware, a complete redesign of the software was done. This comprises the node software, the data forwarder software on the WSN base station, the database connectivity and the data feature extraction and presentation.

This paper reports the experience we had with regard to failures and stability. Reliability is of utmost importance for real-world applications, whose customers are not willing to perform cumbersome maintenance work on the WSN during a monitoring campaign.

A. Reference Deployments

As the field of application of WSN in our institute is monitoring of civil engineering structures, the presented deployment objects are limited to this area.

In the following, the objects are presented shortly, focusing on the positioning of the sensors.

1) Site 1: Johanniskirche Schwäbisch-Gmünd: The Johanniskirche in Schwäbish Gmünd, Germany, is a typical late Romanesque column basilica built in the 13th century. The structure is richly adorned with sculptural ornamentation from the animal and fable world as well as plant embellishments. Water condensation is sometimes seen running down the walls of the clerestory, damaging the paintings. Monitoring moisture in the structure aims to mitigate problems and prevent future damage due to humidity and the resulting condensation. Sensors were positioned at different height levels, predominantly near windows (see Fig. 1 and [4]).

2) Site 2: Neckartalbrücke A6: The Neckartalbrücke near Neckarsulm, Germany, is the longest motorway bridge in the state of Baden-Württemberg. Along the 1350 m total length, different construction principles were used. The section across the river Neckar is a steel frame construction, while the longer section – crossing the Neckar meadows – is a prestressed concrete construction (see Fig. 2). At each construction type, a small WSN system is deployed. For two years, climatic conditions are recorded and cross-checked with the emergence of lane grooves, to prevent the fast erosion of the asphalt layer of the lanes that occurred in the past.

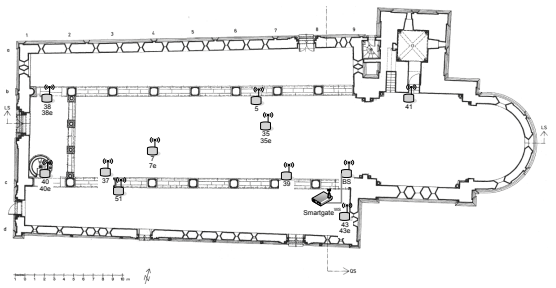


Fig. 1. Top view of Johanniskirche, Schwäbisch Gmünd

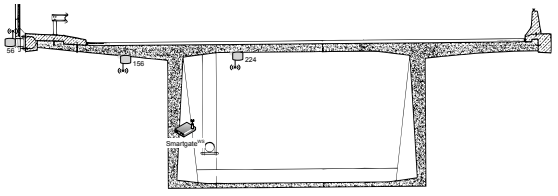


Fig. 2. Section of prestressed concrete bridge with sensors depicted

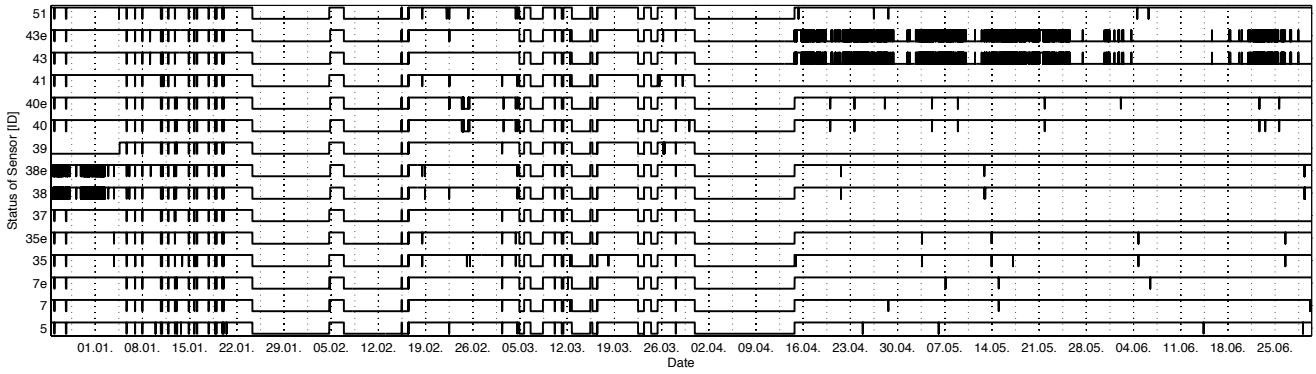


Fig. 3. Uptime status of sensors over time; site 1

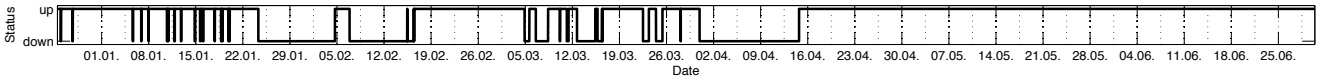


Fig. 4. System uptime status over time; site 1

3) Site 3: Dam "Kleine Kinzig": Situated in the Middle Black Forest, the reservoir dam "Kleine Kinzig" is used for the generation of electrical power, for flood prevention and for the withdrawal of potable water. A service and withdrawal tower in the middle of the reservoir is long-term-monitored for crack openings.

4) Site 4: Schönbrunn Castle Chapel: Schönbrunn Castle, Vienna, Austria, is one of the most important cultural objects of Austria, built in the 17th century and known as a world cultural heritage site. It includes a beautifully decorated Roman-Catholic chapel, situated in a northwestern corner of the castle and is therefore exposed to wind and rain. Problems arise from moisture rising in the wall, which attacks the marble decorations. With the help of the WSN, the condition with respect to temperature, air-flow, air humidity and wall humidity is monitored for more than one year.

II. Survey

A. Survey Description

For each of the presented installation sites, recorded data was evaluated separately. For the extraction of failure rates, the type of sensor was ignored. Only the time stamps were evaluated. The evaluation was done for each sensor individually, where a "high" signal represents a sensor that is operational, and a "low" signal where the sensor has to be assumed unoperational since a data packet was not received. It is generated by evaluation of the measurement points in time, by checking if a scheduled – and hence expected – measurement point is missing. The resulting graph plot is called the uptime graph of this sensor.

The cumulated time where the sensor was operational in relation to the total campaign time is called the uptime ratio.

The system uptime is defined by the time where no data was received from any sensor of the network.

B. Failure Rates

Failures arise from a multitude of reasons. Prominent ones that happened during our test installations are mentioned shortly here:

- bad wide area / mobile network connection
- modem hangup
- freezing base station, either OS or application SW
- mains disturbance
- reboot failures
- faulty hardware.

Bad mobile network connection was seldomly an issue. If they do, they are transient except for where the base station is located at a position with very low or no signal strength. However, modem hangups are often related to a bad connection. The modem has difficulties to reconnect to the network. These difficulties can be permanent and were attributed to a bad modem driver or a bad reconnection script. In our deployment, troubles of that kind occurred with both Windows and Linux operating systems. In rare cases the operating system fully froze; It remained unclear if again the modem driver was involved or if the data forwarding application caused the problem. Mains failures are a frequent cause of system failure in any deployment where the base station is connected directly to a mains supply without a battery backup emergency supply.

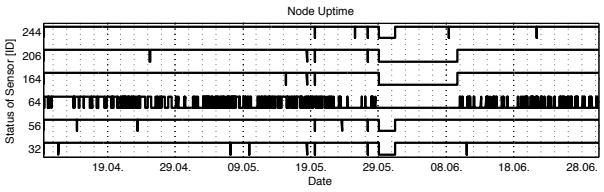


Fig. 5. Uptime status of sensors over time; site 2

All of the mentioned failures are subsumed as system failures.

The calculated system uptime rate is found in Table I. While these rates seem catastrophic at first sight, one has to keep in mind, that the data is unvarnished and results from the first steps of a new sensor network development. In comparison to our institute's first system, many system components were exchanged and new software and hardware was tested. Note, that for ease of evaluation only temperature and humidity data type were evaluated for all sites, even though other sensors were installed on some of the sites.

TABLE I
SYSTEM UPTIME OF THE PRESENTED DEPLOYMENT SITES

Site	System uptime
Johanniskirche	76.1 %
Neckartalbrücke 1	96.9 %
Neckartalbrücke 2	85.7 %
Kleine Kinzig	32.2 %
Schönbrunn	61.8 %

Opposed to the system failures are RF transmission failures that affect single nodes only. These are explained in the following. Fig. 3 shows the uptime graph of each sensor in a long-term deployment at site 1. Depicted here is a period of about half a year. Clearly visible is the missing reception from node 39 before it was moved from behind a column. Sensors 38/38e (attached to the same node) have frequent dropouts in the first time. The cause for that might be that they were located behind a scaffolding, reducing the field strength. Sensors 43/43e (attached to the same node again) show a high dropout rate because they are behind a stone wall of approximately 1 m thickness, through which a hole is levered out. Sensors 43/43e only showed a bad reception of radio packets after they were moved a few centimetres during a site visit. Fig. 4 gives the total system uptime graph of site 1. It can be seen, that after the last maintenance update, where the modem equipement was exchanged, the stability of the system is 100 %.

Fig. 5 shows exemplarily some sensors of site 2. Here, sensors 32, 56 and 244 are connected to one of the base stations, while sensors 64, 164 and 206 are connected to a second one. On 29 May, there was a mains blackout, affecting both systems. While one of the systems came up again automatically, the other one did not reboot fully due to a misconfiguration. It was restarted manually on 9 June. Sensor 64 shows a bad RF connection. After a recent maintenance visit at the bridge, two inductances at the antenna outlet of the sensor node's RF module were found to be burned. This might be either due to an unnoticed error during the soldering process or it was surmised that nearby lightning might have induced a short-time high voltage. After replacement the RF transmission was stable (not shown in the graph).

Due to the concentrated composition of sensors at two distinct and remote locations, a multi-hop network with a single base station would not have been economic.

TABLE II
SENSOR NODE POSITION RELATIVE TO THE BASE STATION AND RESULTING DISTANCE D FROM THE BASE STATION FOR SITE 1

Node	X	Y	Z	D	LOS	Remark
5	-9.1	8.4	0.0	12.4	✓	
7	-20.0	2.8	7.1	21.4	x	Wood ceiling
35	-8.3	5.7	7.1	12.4	x	Wood ceiling
37	-24.5	0.0	0.0	24.5	✓	
38	-31.3	8.2	3.1	32.6	✓	Scaffolding
39	-6.9	-1.0	-6.8	9.7	x	
40	-31.3	0.0	3.1	31.5	x	Scaffolding
41	5.4	8.4	0.6	10.0	✓	
43	0.7	-3.7	-0.4	3.8	x	Stone wall with hole
51	-23.8	-1.0	-6.8	24.8	x	

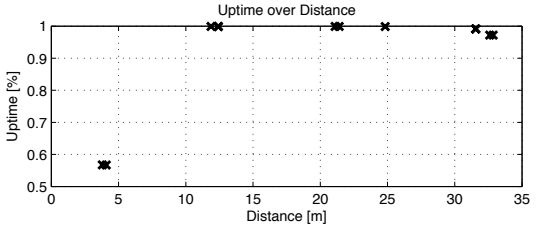


Fig. 6. Uptime rate over distance; site 1

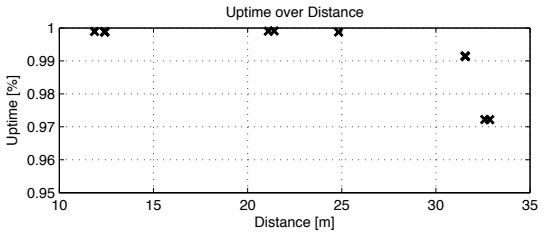


Fig. 7. Uptime rate over distance; site 1; zoomed in

C. Distance based failures

Table II shows the distances of each sensor node to the base station for site 1. The distance is an approximation; one decimal is given, however, the accuracy is not in the 10 cm range, but can vary. Column "LOS" states if line-of-sight conditions are met between sensor node and base station. In the remark column special conditions are indicated. Where no line-of-sight condition is fulfilled but no obstacle is given in the remark section, the sensor node position is behind a corner or in a niche of the stone church structure.

Fig. 6 gives the uptime ratio over the distance for the sensor nodes. Note, that the uptime ratio is corrected by the system downtime, considering only the times where the base station was operational. Sensor nodes with IDs 37, 39 and 41 are not depicted since they were removed from the site during the campaign. Fig. 7 focuses on the range 95 % to 100 % of the same data. Both figures do not surprise, as Fig. 7 just shows the influence of the larger distance on the radio transmission success rate. The outlier in Fig. 6 at less than 4 m distance is explained by the massive lithic church wall which has a small hole only, thus disturbing the radio transmission. The wooden ceiling of the attic has no effect. The influence of

a huge scaffolding, filling the whole nave, which is used for restoration works is unclear. The slightly decreased success rate might be caused by either the scaffolding or by the increased distance.

III. Conclusion

While for some applications, multi-hop-routing might be an optimal solution, for the applications requested by the customers of our system, a star topology was fully adequate. For that reason, software complexity was reduced with a gain in stability. What is gained in a basic transmission protocol is a huge payoff in terms of power consumption. The advantage of easier deployments with enabled multi-hop by not being forced to shift sensors to an optimal position can be out-done by not being forced to replace the battery during the monitoring campaign.

Failure rates were reported in the paper, divided by system components. It was shown, that one of the prime issues is the wide area mobile connection link, not the WSN radio transmission. Many of the problems that occurred were based on the misfunctioning mobile connection drivers or to instable hardware systems.

From a system perspective, it might be necessary to improve first on the base station stability as a single point of failure with many causes, before optimizing transmission protocols. Larger deployments could be accomplished with coupled base stations, similar to installation site 2.

To further enhance stability in real-world applications, a more precise counting of faults will be done in future and an incident database will be established, offering exact statistics and revealing central points for improvements.

Acknowledgment

The authors would like to thank the whole team involved in the development of the system at the institute and would like to thank the EU for their funding contribution.

References

[1] Sustainable Bridges, http://www.sustainablebridges.net/, 2008, date of last access: 2010-07-28

[2] Smart Monitoring of Historic Structures, http://www.smoohs.eu/, 2010, date of last access: 2010-07-28

[3] S. A. Bachmaier, *A Wireless Monitoring System for SHM of Historic Structures. Advantages, Challenges and Applications*, 1st WTA Int. PhD Symposium, Leuven, Belgium, 2009

[4] M. Krüger et al., *Wireless Monitoring Of The Johanniskirche In Schwäbisch Gmünd*, 3rd Int. Workshop on Civil Structural Health Monitoring, Ottawa, Canada, 2010

Deployment of Wireless Sensor Networks in Logistics – Potential, Requirements, and a Testbed

Sebastian Zöller, Andreas Reinhardt, Marek Meyer, Ralf Steinmetz
Multimedia Communications Lab
Technische Universität Darmstadt
Rundeturmstr. 10, 64283 Darmstadt, Germany
{sebastian.zoeller, andreas.reinhardt, marek.meyer, ralf.steinmetz}@kom.tu-darmstadt.de

***Abstract*—With the growing demand for extensive monitoring of transport processes in logistics and approaches for an event-based management of logistics processes, wireless sensor network technology has become a promising technology for this domain. In the first part of this paper, we describe such application possibilities for wireless sensor networks in logistics and focus on supply chain event management as one particularly promising application area, which is often neglected. Afterwards, we present first findings regarding the requirements to be considered for efficient application of wireless sensor networks in logistics. We especially differentiate the design decisions into decisions for the design-time and the run-time of a wireless sensor network deployment in the logistics domain. As such deployments can hardly be tested in real-life scenarios due to organizational and cost reasons, we have built a small-scale testbed. This testbed is presented in the last section of this paper.**

I. INTRODUCTION

Wireless sensor nodes (*motes*) and wireless sensor networks (*WSNs*) offer a variety of capabilities, which make their deployment very promising for several application areas (cf. e.g. [1], [2]), with logistics being one of them, as outlined in Section II. Logistics processes in general and supply chain event management (*SCEM*) in particular can significantly benefit from the sensing and communication possibilities of WSNs. For example, environmental parameters influencing the conditions of transported goods, like tilt, shock, humidity or temperature, can be monitored during the transport process. In case critical values are detected an alarm message with the corresponding event data can be transmitted. Thus, with a deployed WSN such before mentioned events can be detected early and directly at their point of origin during the transport. Additionally, a corresponding notification of relevant decision makers becomes possible using the communication capabilities of the deployed WSN.

For a beneficial exploitation of the existing possibilities, several requirements have to be considered. Consequently, we have examined four requirement categories for the use of WSNs in logistics, which are presented in Section III. These requirements influence criteria concerning the initial design of a WSN in logistics (design-time), as well as decisions concerning the concrete operation of a WSN in logistics during run-time (Section IV). The evaluation of corresponding solutions can hardly take place during normal operations of a freight carrier due to organizational reasons and cost considerations. Therefore, we developed and installed a testbed at the Multimedia Communications Lab (KOM) at Technische Universität Darmstadt (TUD). The testbed can be used for evaluation purposes besides simulation to incorporate real world problems and factors of influence not modelled in simulation tools (cf. e.g. [3]).

II. WSNS IN LOGISTICS – POTENTIAL USE

Several application possibilities for WSNs in the domain of logistics have already been identified. Some initial application possibilities in the context of storage logistics have been described [4], but most often a monitoring of transport processes in the context of transportation logistics is envisioned. Naturally in this context, cold chain monitoring and food logistics are a main focus [5], [6]. One example is the intelligent container [7]. Jedermann et al. use a distributed platform of interacting software agents in combination with a processor module, an RFID system and a WSN deployed in a container. With this system, they want to achieve an autonomous control of transport processes.

We expect SCEM as one particular promising application area for WSNs in the domain of logistics. SCEM can be understood as a management concept as well as a (software) system supporting this management concept [8]. The focus is laid on the detection of so-called events. In this context, events are understood as essential state changes for certain addressees [9]. These events constitute the basis for the management of the supply chain. Their occurrence indicates the requirement for a management action. Thus, a management concept is implemented which leans on the concept of management-by-exception. This management concept needs to be supported by a corresponding (software) system, hence leading to the (software) system perspective of SCEM. SCEM incorporates the five functions ‘monitor’, ‘notify’, ‘simulate’, ‘control’ and ‘measure’ [8], which are executed in this sequence (Fig. 1).

With the sensing, processing and data transmission capabilities of WSNs, we expect that the monitor function, the notify function and the measure function can substantially be supported. The sensing units of motes deployed in a container or a truck’s load area can monitor environmental parameters critical for the condition of transported goods. On this basis, the processing units can execute target-performance comparisons to detect events, e.g. in the form of violation of predefined thresholds. Thus, a significant support of the monitor function can be reached. In case an event is detected, the corresponding information can be transferred through the

WSN and appropriate gateways to responsible decision makers (cf. Section IV), realizing the notify function. Finally, with the available storage capacity on the motes in the WSN, a history of measured environmental parameters and events can be preserved. These can be used as performance indicators in the sense of the measure function to facilitate an assessment of the monitored transport process.

III. Requirements for the Use of WSNs in Logistics

In Section II, we have presented possibilities for the use of WSNs in the domain of logistics. To realize the described potential inherent by this technology, several requirements have to be considered. As these requirements have quite different origins, we distinguish four different categories of requirements:

- Technological Requirements: Comprises properties and constraints of the applied technology, e.g. energy constraints of WSNs.
- Economical and organizational requirements: Comprises economical constraints and potential needs for the integration in an existing infrastructure, e.g. cost-benefit ratio for deployment of WSNs.
- Regulatory requirements: Comprises constraints by law and standardization bodies, e.g. usable frequency bands for transmission.
- Logistics market specific requirements: Comprises properties and constraints of the application domain, e.g. massive cost pressure.

Additionally, interdependencies and conflicting goals between these requirement categories exist. For example, a redundant deployment of motes is preferable as a consequence of technological requirements to ensure functionality despite individual mote failures. But, this implies higher costs, conflicting with logistics market specific requirements.

As we have seen in Section II, enhanced information availability can be exploited in several ways and can lead to significant benefits. But this enhanced information availability realized by WSNs does not come for free. Therefore, and especially against the background of the massive cost pressure in the logistics market, a sufficient cost-benefit ratio must be ensured as part of the logistics market specific requirements. Consequently, a thorough and detailed investigation of the economical value of a specific WSN deployment in a logistics context should be mandatory. Unfortunately, most often a technological view focussing on technological requirements only is chosen.

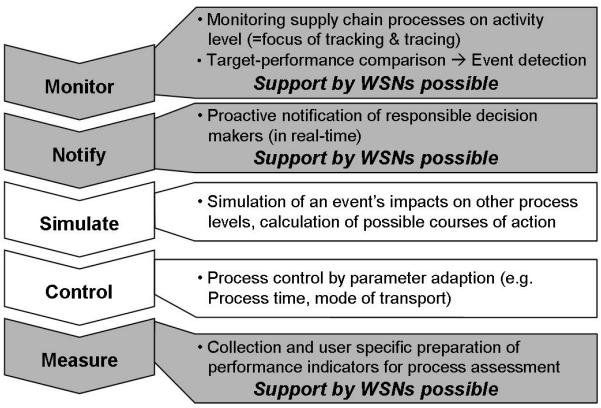


Fig. 1. Functional components and possible WSN support of SCEM systems (based on [8])

IV. Design-Time and Run-Time Design Decisions

Designing a WSN deployment several design questions have to be answered and corresponding decisions have to be taken. These design decisions can be divided in decisions concerning the initial layout of the WSN (design-time) and decisions concerning the operation of the deployed WSN (run-time). In the following, we will focus on decisions for the design-time and cover run-time decisions just shortly.

A. *Design-Time Decisions*

With a WSN being formed by collaborating autonomous motes and understanding that in the described application context the WSN-data is needed at several end users respectively their systems, amongst other decisions, fundamental selections of platform, number of motes, location of motes, and connectivity must be made.

All these design decisions have to be taken against the specific application background, in our case transport processes in the domain of transportation logistics, and taking into account the requirement categories described in Section III. So, for example to answer the question which motes should be used, there has to be a decision which environmental parameters have to be monitored, how much money can be spent and so on. As no generally applicable solution exists, we do not provide a generic solution at this point. Instead, we focus on the network connectivity next and address the question on how a connection between a WSN and an end user, in our case the decision maker responsible for reaction to an occurred event during the transport, can be established.

We have identified two basic alternatives to establish a connection between a WSN deployed in a container on a truck or a truck's load area and a decision maker responsible for the corresponding transport process: The connection can be established by using devices already available in the truck and able to establish a long-distance connection to the end user's system (Fig. 2) or by employing dedicated devices just for connecting the In-Truck-WSN with the decision maker's system (Fig. 3).

It is very probable that the truck driver at least possesses a simple cell phone or already a smartphone. Otherwise, he can easily be equipped with one. Consequently, it can be assumed that such a device is available to be used to connect the WSN to the end user's system. Besides, many trucks carry a so-called On-Board-Unit (OBU), which is for example needed for toll accounting. These are two different devices which could be used to establish a connection between WSN and the end user.

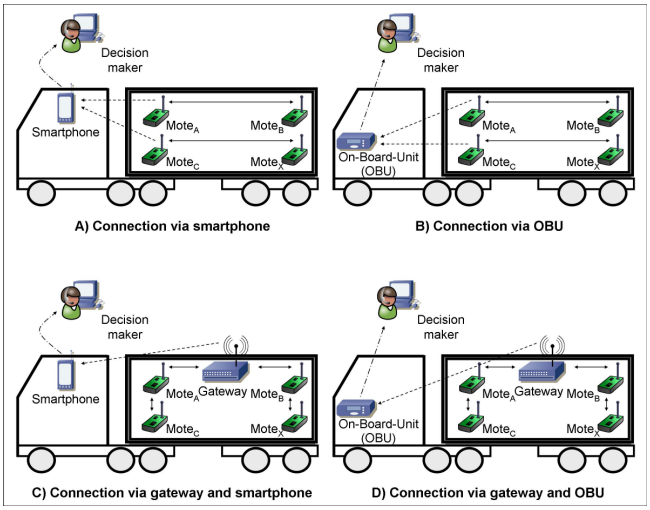


Fig. 2. Possibilities for connection of an In-Truck-WSN to end users via existing devices

Besides using such already existing devices, another option would be the deployment of dedicated, specialized gateways, solely used for the connection between WSN and end user.

Regarding the economical and organizational requirements as well as the logistics market specific requirements with their massive cost pressures and the corresponding cost-efficiency needed, a usage of already existing devices seems very promising. This would imply the use of a smartphone or the OBU. In addition to the capability to establish a long-range connection to the end user, a smartphone would feature a well-known and easy-to-use interface to the truck driver. This interface would allow giving the driver instructions directly in case critical parameters are detected. So, smartphones seem to be very promising devices for the connection of an In-Truck-WSN to an end user's system. Furthermore, a smartphone would provide additional computing and storage resources, as well. These could be used for example for aggregation of data received from the WSN. As a consequence, we think that in future research possibilities to connect In-Truck-WSNs with smartphones should be pursued intensely.

B. Run-Time Decisions

Taking the requirement categories of Section III as a basis, two criteria for the run-time of a WSN deployed in logistics can be identified as most important:

- Energy-efficient operation as a result of technological requirements

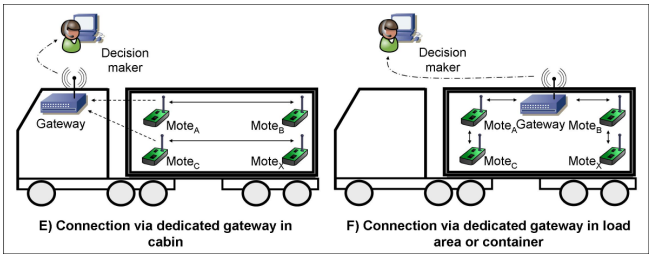


Fig. 3. Possibilities for connection of an In-Truck-WSN to end users via specialized gateways

- Cost-efficient operation as a result of logistics market specific requirements

We address these criteria with the concept of so-called *transmission-relevant events*. In the domain of logistics and especially in the context of SCEM events are understood as essential state changes for certain addressees (cf. Section II). As transmission-relevant events, we specify events, which possess an information value higher than their transmission costs. Thereby, our understanding of transmission costs comprises costs in terms of energy and in terms of money. So, in case an event is detected by a mote, it uses current context data and further information to decide whether it is a transmission-relevant event or not and only transmits data of transmission-relevant events. This way, we expect to significantly enhance both energy-efficient operation and cost-efficient operation of WSNs deployed in transport processes.

V. TESTBED DESIGN

Solutions and approaches in the context of WSNs in transport processes cannot be tested during regular operations of a freight carrier due to organizational and cost reasons. Hence, we have developed a testbed at KOM at TUD, which can be used besides simulation to test solutions and approaches in a more realistic setting, incorporating some real world problems and factors of influence not modelled in simulations.

To support intermodal transportation scenarios, small-scaled models of different means of transport, like containerships, trucks and trains, as well as containers have been deployed in a modelled landscape with various transport routes featuring different characteristics (Fig. 4). Thus, we currently provide a transportation scenario starting with the arrival of a container at a port which is then transported either only by truck or by train and truck to a warehouse as the point of destination. Currently, we have equipped the model containers with SunSPOTs as they are easy to program, provide all sensor capabilities needed for our purposes, are well supported and documented, possess a wide dissemination in the community, and are relatively cheap. Naturally, if the detailed analysis of the design questions mentioned in Section IV or gathered experience should yield that SunSPOTs are not suited anymore, we could switch to other mote platforms as well. Environmental parameters relevant for a wide variety of transported goods, e.g. high value consumer electronics like plasma screens, pharmaceuticals and medicine like swine flu vaccine or food like bananas, are tilt, shock and temperature. Thus, we currently test their influence and provide real-time monitoring of these critical parameters with the deployed SunSPOT-infrastructure. Changes of these parameters are modelled by the influence of the characteristics of different transport routes. For example, we deployed a model bridge, which causes tilt changes when used in the context of a container transport by truck. Furthermore, we have deployed model roads with potholes, causing sudden shocks. And finally, we have deployed an infrared lamp which can be used to change the temperature of the interior of the container. Additionally, we use an RFID infrastructure with six RFID-readers installed at the different transshipment points and the warehouse as point of destination combined with the corresponding tags applied to the model container. This way,

tracking and tracing of the model container as it is transported from the port to the warehouse is implemented, as well.

Currently, we focus on run-time behaviour with testing possibilities for real-time monitoring of the transport process of our model container through our test scenario. Furthermore, possibilities for event detection and transmission with WSNs in logistics are analyzed. In the future, we plan to integrate some possibilities to evaluate design-time decisions as well, for example by integrating mobile devices like smartphones to test possibilities for their connection with a WSN (cf. Section IV).

VI. Related Work

The application of WSNs in the logistics domain is a main research aspect of Jedermann and his colleagues involved in the Collaborative Research Centre 637 "Autonomous Cooperating Logistic Processes"[1] located at the University of Bremen. Their focus is on developing an intelligent container, equipped with a processor module, an RFID system and a WSN. Using this intelligent container and software agents they work on realizing autonomously controlled transport processes with a distinct focus on food and cold chain logistics. Ruiz-Garcia et al. are focussing their work on WSNs in logistics on food and cold chain logistics, too [5], [10]. In this context, they research energy-efficient design of motes and communication infrastructures of WSNs. Evers et al. have as well chosen logistics as application scenario for their work in WSNs [11]. They too address energy efficiency, but regarding reprogramming of motes for which they provide an energy-efficient, secure, flexible and dynamic solution.

The described research work mostly adopts a strong technological view, primarily addressing technological requirements. Logistics market specific requirements are rarely considered or only on a rather basic, very abstract level.

VII. Conclusions and Outlook

We have identified the potential use for WSNs in the context of supply chain event management as significant and very promising. To realize this potential with the deployment of a WSN in the logistics domain, requirements of the categories technological requirements, economical and organizational requirements, regulatory requirements and logistics market specific requirements as well as their interdependencies have to be considered. These requirements influence both decisions concerning the design-time of a WSN and concerning the intended run-time behaviour of a WSN. Several of the design decisions to be taken have been described. As one of these, possibilities to connect an In-Truck-WSN to end user systems have been investigated exemplary. As a result, we identified the connection via smartphone as a very appealing approach, which should be investigated in more detail in future work. Finally, we have described our testbed and presented it as one possibility to evaluate approaches in the context of WSN deployments in logistics processes.

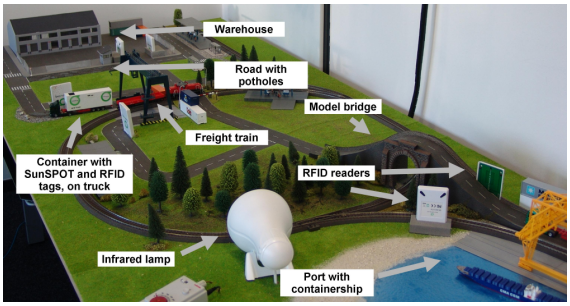


Fig. 4. Model of application scenario used as testbed

In the next steps, the described design questions concerning design-time and run-time of a WSN to be deployed in a logistics process have to be analyzed in more detail. This has to be done considering the background of the mentioned requirement categories and clearly with a broadened focus not just addressing technological requirements, but explicitly taking into account logistics market specific requirements, as well. The resulting solutions will be evaluated in our testbed.

[1] http://www.sfb637.uni-bremen.de

References

[1] K. Römer and F. Mattern, "The design space of wireless sensor networks," *IEEE Wireless Communications,* vol. 11, no. 6, pp. 54–61, December 2004.

[2] H. Karl and A. Willig, *Protocols and Architectures for Wireless Sensor Networks*. Chichester: Wiley, 2007.

[3] K. Langendoen, A. Baggio, and O. Visser, "Murphy loves potatoes," in *Proc. IPDPS, 2006*.

[4] S. Haller and Z. Nochta, "Kooperation zwischen intelligenten Gütern," *Industrie Management*, vol. 22, no. 3, pp. 53–56, 2006.

[5] L. Ruiz-Garcia, P. Barreiro, J. Rodriguez-Bermejo, and J. I. Robla, "Review. Monitoring the intermodal, refrigerated transport of fruit using sensor networks," *Spanish Journal of Agricultural Research*, vol. 5, no. 2, pp. 142–156, 2007.

[6] R. Jedermann and W. Lang, "The minimum number of sensors – Interpolation of spatial temperature profiles in chilled transports," in *Proc. EWSN*, 2009, pp. 232–246.

[7] R. Jedermann, J. D. Gehrke, M. Lorenz, O. Herzog, and W. Lang, "Realisierung lokaler Selbststeuerung in Echtzeit: Der Übergang zum intelligenten Container," in *Wissenschaft und Praxis im Dialog*, H.-C. Pfohl and T. Wimmer, Eds. Hamburg: Deutscher Verkehrs-Verlag, 2006, pp. 145–166.

[8] T. Placzek, "Potenziale der Verkehrstelematik - zur Abbildung von Transportprozessen im Supply Chain Event Management," *Logistik Management*, vol. 6, no. 4, pp. 34–46, 2004.

[9] W.-R. Bretzke and M. Klett, "Supply Chain Event Management als Entwicklungspotenzial für Logistikdienstleister," in *Supply Chain Management*, H. Beckmann, Ed. Berlin: Springer, 2004.

[10] L. Ruiz-Garcia, P. Barreiro, and J. Robla, "Performance of Zigbee-based wireless sensor nodes for real-time monitoring of fruit logistics," *Journal of Food Engineering*, vol. 87, no. 3, pp. 405–415, 2008.

[11] L. Evers, P. Havinga, and J. Kuper, "Flexible sensor network reprogramming for logistics," in *Proc. MASS*, 2007, pp. 1–4.

[12] R. Jedermann, C. Behrens, R. Laur, and W. Lang, "Intelligent containers and sensor networks - Approaches to apply autonomous cooperation on systems with limited resources," in *Understanding autonomous cooperation and control in logistics*, M. Hülsmann and K. Windt, Eds. Berlin: Springer, 2007, pp. 365–392.

Über die Notwendigkeit einer Integrationsplattform für unterschiedliche Smart Object Technologien

Sebastian Lempert, Alexander Pflaum
Geschäftsfeld Technologien
Zentrum für Intelligente Objekte ZIO
Fraunhofer-Arbeitsgruppe für Supply Chain Services SCS
Fraunhofer-Institut für Integrierte Schaltungen IIS
Dr.-Mack-Straße 81
90762 Fürth
{vorname.nachname}@scs.fraunhofer.de

Abstract—**Obwohl die Einbindung von unterschiedlichen Smart Object Technologien wie WSN, RFID und RTLS in bestehende Infrastrukturen für die Praxis von zentraler Bedeutung ist, wird dieser Umstand von der Forschung häufig ignoriert. Weiterhin ist zu beobachten, dass Anbieter von Middleware-Produkten den Trend der Verschmelzung unterschiedlicher Smart Object Technologien ignorieren und sich bei der Anbindung dieser Technologien an bestehende IT-Systeme von Unternehmen auf eine Auswahl dieser Technologien beschränken. In dem vorliegenden Papier wird zunächst das Konzept einer Integrationsplattform für Smart Object Technologien vorgestellt, welche die genannten Probleme adressiert. Weiterhin stellt dieser Beitrag die aus der Praxis abgeleiteten Anforderungen an eine solche Integrationsplattform für Smart Object Technologien vor, auf deren Basis der Integrationsaufwand für diese Technologien erheblich reduziert werden kann.**

I. Einleitung und Motivation

Anforderungen an drahtlose Sensornetze werden von Industrie und Forschung häufig unterschiedlich bewertet. Während in der Sensornetzforschung Integrationsaspekte mit bestehender Infrastruktur häufig außen vorgelassen werden, sieht die Realität oft anders aus: in der Praxis stellt die Anbindung eines drahtlosen Sensornetzes an eine existierende Unternehmens-IT einen bedeutsamen Kostentreiber dar und ist von zentraler Bedeutung [1]. Sensornetze können nur dann gewinnbringend eingesetzt werden, wenn diese mit möglichst geringem Aufwand in die bestehende IT-Infrastruktur der Unternehmen eingebunden werden können.

Des Weiteren werden drahtlose Sensornetze in der Forschung häufig für sich allein betrachtet, obwohl eine zunehmende Verschmelzung mit anderen Smart Object Technologien wie Identifikations-, Kommunikations- und Ortungstechnologien zu beobachten ist. Als Stellvertreter der Identifikationstechnologien sei hier die RFID-Technologie [2] erwähnt, wo bei der noch nicht abgeschlossenen Standardisierung aktiver RFID-Tags zu erkennen ist, dass diese zukünftig nicht nur mit eigener Spannungsversorgung und zusätzlicher Sensorik versehen sind, sondern möglicherweise auch Ad-hoc-Kommunikation zwischen unterschiedlichen Tags ohne Einbeziehung eines RFID-Readers beherrschen können [3]. Die zunehmende Verschmelzung von Smart Object Technologien lässt sich aber auch anhand der so genannten Real-Time Locating Systems (RTLS) [4] erkennen, da eine Positionsbestimmung auch auf Basis von RFID [5] und drahtlosen Sensornetzen [6] möglich ist.

Vor diesem Hintergrund weist das vorliegende Papier auf die Notwendigkeit einer Integrationsplattform für unterschiedliche Smart Object Technologien hin, welche diese Technologien über eine gemeinsame Hardware-Abstraktionsschicht vereint, das Zusammenspiel mit der vorhandenen Unternehmens-IT regelt und somit den Integrationsaufwand erheblich reduziert.

Der weitere Aufbau dieses Beitrags gestaltet sich wie folgt: Abschnitt II schärft das Verständnis des Begriffs einer Integrationsplattform für Smart Object Technologien durch eine Gegenüberstellung der Begriffe Middleware und Integrationsplattform sowie durch eine darauf aufbauende Begriffsdefinition. Auf dieser Basis nimmt Abschnitt III mit einer kurzen Betrachtung existierender Middlewares für drahtlose Sensornetze, RFID und RTLS eine Einordnung in das fachliche Umfeld vor. Darauf aufbauend werden in Abschnitt IV die aus Sicht der Autoren wichtigsten Anforderungen an eine Integrationsplattform für unterschiedliche Smart Object Technologien vorgestellt. Abschnitt V schließt diesen Beitrag ab, indem zuerst eine rückblickende Zusammenfassung und darauf aufbauend ein Ausblick auf zukünftige Entwicklungen und den Fortgang dieser Arbeit gegeben wird.

II. Begriffsabgrenzung

A. *Middleware*

Eine klare Definition des Begriffs Middleware fällt schwer, da der Begriff in der Literatur häufig in unterschiedlichen Kontexten verwendet wird. Zum anderen ist die Bedeutung des Begriffs durch die häufige Verwendung für Marketingzwecke kommerzieller Hersteller von Produkten mit unterschiedlichem Kontext verwässert worden und hat eine Art Buzzword-Charakter inne. Die Autoren dieses Papiers erachten es daher als notwendig, bei der Verwendung des Begriffs Middleware zu spezifizieren, was eigentlich genau damit gemeint ist. Eine eingängige allgemeine Definition des Begriffs Middleware lautet frei übersetzt: Middleware kann im weitesten Sinne als

Oberbegriff für Software verstanden werden, welche die Interaktion zwischen unterschiedlichen Software-Produkten ermöglicht [7].

B. *Integrationsplattform*

Darauf aufbauend bezeichnet eine Integrationsplattform ein Middleware-Produkt oder eine Kombination solcher Produkte, die es ermöglichen, verschiedene Applikationen im Sinne der Enterprise Application Integration (EAI) entlang der Wertschöpfungskette prozessorientiert zu verbinden [8].

C. *Integrationsplattform für Smart Object Technologien*

Im Kontext von Smart Object Technologien wie RFID, WSN und RTLS wird darauf aufbauend der Begriff Integrationsplattform am Zentrum für Intelligente Objekte ZIO [9] wie folgt definiert: Eine Integrationsplattform für Smart Object Technologien stellt eine Integrationsplattform dar, welche diese Technologien über eine gemeinsame Hardware-Abstraktionsschicht vereint, das Zusammenspiel mit der vorhandenen IT sowohl unternehmensintern als auch unternehmensübergreifend regelt und somit den Aufwand für die Anbindung dieser Technologien erheblich reduziert. Zudem sollte eine solche Integrationsplattform aus Sicht der Autoren die in Abschnitt IV genannten Anforderungen erfüllen.

III. Verwandte Arbeiten

A. *WSN-Middlewares*

Insgesamt wurden im Zuge der Recherchen zu dem vorliegenden Papier über 20 Studien aus dem Zeitraum 2002 bis 2010 ausgewertet, die über 120 verschiedene WSN-Middlewares und zugehörige Anforderungen und Entwurfsprinzipien auflisten. Erstmals wurde mit [10] ein Papier dieser Art vorgestellt, welches in [11] weiter verfeinert wird und eine erste Klassifizierung von WSN-Middlewares vornimmt. Weitere erwähnenswerte Studien lassen sich den folgenden Kategorien für WSN-Middlewares zuordnen, welche sich weitestgehend an den Kategorien von [12] orientieren: Betriebssysteme [13] und virtuelle Maschinen [14], Programmierabstraktionen und –unterstützung [15, 16] und Middlewares [12, 17].

Den in diesem Abschnitt vorgestellten Studien ist gemein, dass allesamt aus dem Forschungsumfeld stammen, welche wiederum ausschließlich WSN-Middlewares aus dem Forschungsumfeld untersuchen. Weiterhin fällt auf, dass ausschließlich in [11] Mechanismen zur Einbindung eines drahtlosen Sensornetzes in die bestehende Unternehmens-IT als wichtiger Bestandteil einer WSN-Middleware angesehen werden, ohne jedoch Produkte zu listen, die diese Mechanismen bereit stellen können.

B. *RFID-Middlewares*

RFID-Middleware stellt den Software-Teil innerhalb eines RFID-Stacks dar, welcher die Konvertierung, Filterung und Aggregation von Tag-Daten, sowie die Ereignisverarbeitung und Anwendung von Geschäftsregeln übernimmt. Weiterhin ist RFID-Middleware zuständig für die Verwaltung von RFID-Geräten und das Beschreiben von RFID-Tags mit Daten [18].

Eine frühe Übersicht über verfügbare RFID-Middlewares, charakteristische Funktionalitäten einer RFID-Middleware sowie eine Marktübersicht inkl. Klassifizierung der Anbieter geben [19] und [20]. Eine aktuelle und sehr umfangreiche Auflistung von Funktionalitäten einer RFID-Middleware, eine umfassende Marktübersicht sowie ein Überblick über die wichtigsten Standards gibt [18]. Hier ist besonders erwähnenswert, dass die Forderung nach der Einbindung einer Business Rule Engine (BRE) [21] sowie einer Lösung für das Complex Event Processing (CEP) [22] laut wird, um mit diesen Technologien die Überwachung von Geschäftsprozessen in Echtzeit und die frühzeitige Erzeugung von Fehler- und Warnmeldungen bei Eintritt kritischer Ereignisse zu ermöglichen.

Allen genannten Übersichten ist gemein, dass die Anbindung an eine existierende Unternehmens-IT als wichtig erachtet wird und ausschließlich kommerzielle RFID-Middlewares betrachtet werden. Abschließend ist anzumerken, dass zwar einige wenige Produkte genannt werden, die explizit die Unterstützung von drahtlosen Sensornetzen und RTLS anbieten, jedoch ist fraglich, ob es sich dabei um eine vollwertige Unterstützung dieser Technologien handelt.

C. *RTLS-Middlewares*

In [4] wird RTLS-Middleware zwar als Software beschrieben, welche zwischen den Komponenten der jeweils eingesetzten RTLS-Technologie (Tags, Bezugspunkte, Location Engine) und der Anwendungssoftware angesiedelt ist, jedoch konnten während der Recherchen zu diesem Beitrag weder Informationen zu einem entsprechenden Produkt noch zu einer Übersicht über unterschiedliche Produkte ermittelt werden.

D. *Open-Source-basierte Middlewares mit teilweiser Anbindung von RFID und WSN*

In einer am Zentrum für Intelligente Objekte ZIO durchgeführten, bisher unveröffentlichten Studie wurden über 50 kommerzielle RFID-Middlewares und über 70 Open-Source-Lösungen mit direktem Bezug zu RFID identifiziert, wobei von diesen Open-Source-Lösungen nur knapp über zehn Produkte einer eigenständigen RFID-Middleware entsprechen. Unter Berücksichtigung der Praxistauglichkeit reduziert sich die Zahl auf folgende vier Produkte: GSN [23], Fosstrak (ehemals Accada) [24], ASPIRE [25] und Rifidi [26]. Alle Produkte unterstützen RFID und bis auf Fosstrak zusätzlich auch Sensornetze. Jedoch ist auch hier fraglich, ob es sich dabei um eine vollwertige Unterstützung der jeweils zusätzlichen Smart Object Technologie handelt.

IV. Anforderungen an eine Integrationsplattform für unterschiedliche Smart Object Technologien

A. *Hardware-Abstraktion*

Unterschiedliche Smart Object Technologien müssen über eine gemeinsame Hardware-Abstraktionsschicht (HAL) angebunden werden, welche die Kommunikation mit und die Verwaltung von beliebigen Aktoren und Sensoren ermöglicht, ohne dabei die Funktionalität der jeweiligen Technologie zu

weit einzuschränken. Beispielsweise ist es aus Sicht der Autoren nicht ausreichend, ein Gateway eines Sensornetzes innerhalb der Middleware lediglich als RFID-Reader darzustellen, da so ein Großteil der nutzbaren Funktionalität von Sensornetzen brach liegt. Weiterhin sollte sichergestellt sein, dass für jeden Sensorwert auch Zeit und Ort festgehalten werden: ein Sensorwert ist wertlos ohne die Information wann und wo der Sensorwert gemessen wurde. Zusätzlich sollten Zeitangaben, Ortsangaben und Sensorwerte jeweils über einen Qualitätsindikator verfügen, welcher anzeigt, wie hoch die Genauigkeit der jeweiligen Information unter den gegenwärtigen Umgebungsbedingungen ist.

B. Anbindung von existierender Enterprise Software

Neben einer Web Service Schnittstelle sollten spezielle Konnektoren für besonders häufig in Unternehmen anzutreffende Produktgattungen wie SCM, ERP, WMS und CRM zur Verfügung gestellt werden. Eine geeignete Integrationsplattform sollte zudem zeitgemäße Methoden der Enterprise Application Integration (EAI) unterstützen, um verschiedene Anwendungen entlang der Wertschöpfungskette prozessorientiert verbinden zu können. Die Autoren dieses Papiers sehen insbesondere den Einsatz eines ESB als vielversprechend an [27].

C. Statusmonitor

Eine Integrationsplattform sollte den Status aller über die HAL angeschlossenen Geräte, aber auch den Status der Schnittstellen zur Unternehmens-IT überwachen und zugehörige Informationen geeignet visualisieren. Hierzu zählen der Heartbeat und darüber hinaus gehende Mechanismen, da der Heartbeat allein noch keine Auskunft darüber gibt, ob die antwortende Komponente tatsächlich einwandfrei funktioniert und arbeitet.

D. Standardunterstützung

Für genannten Smart Object Technologien sowie im Umfeld der EAI ist eine Vielzahl von (De-facto-)Standards anzutreffen. Da es unwahrscheinlich erscheint, dass eine Integrationsplattform alle Standards unterstützt, sollte genau überlegt werden, welche Standards als besonders relevant erachtet werden.

E. Event Management

Das Event Management hat die Überwachung von Geschäftsprozessen in Echtzeit und die frühzeitige Erzeugung von Fehler- und Warnmeldungen bei Eintritt kritischer Ereignisse zur Aufgabe. Um ein effizientes Event Management auf Basis von Smart Object Systemen zu ermöglichen, sollte eine BRE in die Integrationsplattform integriert werden [21]. Alternativ bzw. ergänzend könnte auch der Ansatz des CEP herangezogen werden [22].

F. Datenbank und Datenpufferung

Vom Event Management erkannte Ereignisse sollten in einer Datenbank abgelegt und abgefragt werden können. Bei dem Design der Datenbank sollte auf projekt- und. anwendungsszenarioübergreifende Wiederverwendbarkeit geachtet werden. An dieser Stelle könnte das so genannte Entity-Attribute-Value-Modell hilfreich sein, welches zum Einsatz kommt, wenn die Anzahl der Attribute zur Beschreibung einer Entität im Vorfeld nicht vorhersehbar ist. Zudem sollten für den Fall, dass die Verbindung zur Unternehmens-IT zwischenzeitlich ausfällt, Mechanismen zur Pufferung von Ereignissen/Informationen vorhanden sind.

G. Benutzerverwaltung und Sicherheit

Eine Integrationsplattform sollte den Umgang mit Benutzern, Gruppen und Rollen inkl. dazugehöriger Berechtigungen beherrschen. Weiterhin sollten Mechanismen zur Authentifizierung und Autorisierung integriert werden.

H. Update-Mechanismen

Die Verwaltung einer großen Anzahl eingesetzter Smart Objects kann eine zeitraubende Aufgabe sein. Im Laufe der Zeit gilt es beispielsweise die Firmware der eingesetzten Geräte auf einem aktuellen Stand zu halten. Daher ist es wichtig, dass dieser Update-Prozess möglichst automatisiert für gleichartige Geräte erfolgen kann. Zudem sollten Mechanismen vorhanden sein, die ein Update der Integrationsplattform selbst unterstützen.

I. Mapping zwischen Tag und zugeordnetem Objekt

Smart Object Technologien werden in der Regel eingesetzt, um ein Objekt oder eine Person informationstechnisch zu überwachen. Beispielsweise wird im Rahmen des Asset Managements ein RFID-Tag oder ein Sensorknoten an das zu überwachende Asset angebracht. Da eine Vielzahl von Geschäftsregeln darauf angewiesen ist, dass das durch das Tag repräsentierte Objekt selbst bekannt ist, muss eine Integrationsplattform in der Lage sein, ein Mapping zwischen einem Tag und dem zugeordnetem Objekt vorzunehmen.

J. Tracking und Tracing

Der aktuelle Aufenthaltsort (Tracking) sowie die Historie der früheren Aufenthaltsorte eines Objektes (Tracing) müssen gespeichert, abgefragt und visualisiert werden können.

K. Filterung und Aggregation

Nicht alle Informationen, die die verwendete Smart Object Technologie liefert, ist relevant. Eine Integrationsplattform sollte daher Mechanismen vorhalten, welche das Filtern und Aggregieren von Daten ermöglichen.

L. Verteilung von Middleware-Funktionalität auf unterschiedliche Geräteklassen

Die Frage, welche Daten bzw. Funktionalitäten innerhalb einer RFID-Architektur zentral oder dezentral vorgehalten werden sollen, ist Gegenstand vieler Forschungsarbeiten und noch nicht abschließend geklärt. Dezentralisierung auf Basis von Software-Agenten kann bspw. zu skalierbareren Systemen führen, gleichzeitig aber auch zu redundanten Daten führen, da ein Agent gegebenenfalls Daten repliziert vorhalten muss, um Entscheidungen selbständig fällen zu können.

V. Zusammenfassung und Ausblick

Ausgehend von der Feststellung, dass die Einbindung von unterschiedlichen Smart Object Technologien wie WSN, RFID und RTLS in bestehende Infrastrukturen für die Praxis von zentraler Bedeutung ist und dem Umstand, dass ein Trend der Verschmelzung unterschiedlicher Smart Object Technologien zu beobachten ist, wurde die Notwendigkeit einer Integrationsplattform für unterschiedliche Smart Object Technologien aufgezeigt. Zu diesem Zweck wurde zunächst das Verständnis des Begriffs einer Integrationsplattform für Smart Object Technologien durch eine Gegenüberstellung der Begriffe Middleware und Integrationsplattform sowie durch eine darauf aufbauende Begriffsdefinition geschärft. Auf dieser Basis wurde mit einer kurzen Betrachtung existierender Middlewares für drahtlose Sensornetze, RFID und RTLS eine Einordnung in das fachliche Umfeld vorgenommen. Abschließend wurden darauf aufbauend die aus Sicht der Autoren wichtigsten Anforderungen an eine Integrationsplattform für unterschiedliche Smart Object Technologien vorgestellt.

Die vorgestellten Anforderungen an stellen eine erste Näherung an einen umfassenden Anforderungskatalog dar, welcher noch fertiggestellt werden muss. Ausgehend von diesem Anforderungskatalog gilt es eine Referenz-Architektur zu entwerfen, Schritt für Schritt prototypisch umzusetzen und gemeinsam mit Partnern aus Industrie und Forschung zu evaluieren.

Literaturverzeichnis

[1] Falk, Rainer; Niedermeier, Christoph; Hof, Hans-Joachim; Sollacher, Rudolf; Meyer, Ulrike; Vicari, Norbert: From Academia to the Field – Wireless Sensor Networks for Industrial Use. In: FGSN 2008 (Proceedings of the 7. GI-ITG KuVS Fachgespräch Drahtlose Sensornetze, Berlin, Deutschland, 2008-09-25/26). Berlin, Deutschland: Freie Universität Berlin, 2008 (TR B 08-12). - Forschungsbericht

[2] Finkenzeller, K.: RFID Handbuch – Grundlagen und praktische Anwendungen induktiver Funkanlagen, Transponder und kontaktloser Chipkarten. 5., aktualisierte und erweiterte Auflage, München: Carl Hanser Verlag, Oktober 2008. – ISBN 3-446-41200-X

[3] Dienelt, Sven: EPC/RFID und Sensorik – Grundlageninformation – EPC/RFID und Sensorik – Einführung, Einsatzgebiete und Standardisierung. GS1 Germany GmbH / EPCglobal Inc., Juli 2009. - Firmenschrift

[4] Malik, A.: RTLS for Dummies. Hoboken, New Jersey, USA: Wiley Publishing, 2009. – ISBN 978-0-470-39868-5

[5] Mojix Inc. (Hrsg.): Mojix EPC Compliant Real-Time Location System. Los Angeles, Kalifornien, USA: Mojix Inc., 2009. – Firmenschrift

[6] Karl, H.; Willig, A.: Protocols and Architectures for Wireless Sensor Networks / Karl, H.; Willig, A.. Chichester, West Sussex, England: John Wiley & Sons, 2007. – ISBN 978-0-470-51923-3

[7] Defining Technology, Inc.: Middleware Resource Center – What is Middleware? URL: http://middleware.org/whatis.html [Stand 2010-07-06]

[8] Wikipedia, Die Freie Enzyklopädie (Hrsg.): Integrationsplattform. URL: http://de.wikipedia.org/w/index.php?title=Integrationsplattform&oldid=64795310 [Stand 2010-07-06]

[9] Zentrum für Intelligente Objekte ZIO (Hrsg.): Offizielle Webseite. URL: http://www.zio.fraunhofer.de/ [Stand 2010-07-06]

[10] Römer Kay; Kasten, Oliver; Mattern, Friedemann: Middleware Challenges for Wireless Sensor Networks. In: ACM Mobile Computing and Communication Review 6 (2002), Nr. 4, S. 59-61 - ISSN 1559-1662

[11] Römer, Kay: Programming Paradigms and Middleware for Sensor Networks. In: FGSN 2004 (Proceedings of the 2. GI-ITG KuVS Fachgespräch Drahtlose Sensornetze, Karlsruhe, Deutschland, 2004-03-26/27). Karlsruhe, Deutschland: Universität Karlsruhe (TH), 2004. - Forschungsbericht

[12] Yoneki, Eiko; Bacon, Jean: A survey of Wireless Sensor Network technologies - research trends and middleware's role / Computer Laboratory, University of Cambridge. Cambridge: University of Cambridge, September 2005 (Technical Report UCAM-CL-TR-646). - ISSN 1476-2986

[13] Adi Mallikarjuna Reddy V AVU Phani Kumar, D Janakiram, G Ashok Kumar: Operating Systems for Wireless Sensor Networks - A Survey. In: International Journal of Sensor Networks 5 (2009), Nr. 4, S. 236-255. – ISSN 1748-1279

[14] Costa, Nuno; Pereira, António; Serôdio, Carlos: Virtual Machines Applied to WSN's – The state-of-the-art and classification. In: ICSNC (Proceedings of the 2nd International Conference on Systems and Networks Communications, Cap Esterel, Frankreich, 2007-08-25/31). IEEE Computer Society, 2007. – ISBN 0-7695-2938-0, S. 50

[15] Hadim, Salem; Mohamed, Nader: Middleware - Middleware Challenges and Approaches for Wireless Sensor Networks. In: IEEE Distributed Systems Online 7 (2006), Nr. 3. – ISSN 1541-4922

[16] Mottola, Luca; Picco, Gian Pietro: Programming Wireless Sensor Networks: Fundamental Concepts and State of the Art. In: ACM Computing Surveys. – ISSN 0360-0300, bisher unveröffentlicht

[17] Kuorilehto, Mauri: System Level Design Issues in Low-Power Wireless Sensor Networks. Tampere, Tampere University of Technology, Institute of Digital and Computer Systems, Dissertation, 29. April 2008. – ISBN 978-952-15-2006-8

[18] Holloway, Simon: RFID Middleware - From RFID to Sensory Network middleware for the edge. London, England: Bloor Research, Mai 2008. – Forschungsbericht

[19] Leaver, Sharyn; Mendelsohn, Tamara; Spivey Overby, Christine; Yuen, Esther H.: Evaluating RFID Middleware – Picking The Right Solution For Integrating RFID Data Into Business Applications. Cambridge, Massachusetts, USA: Forrester Research, August 2004. - Forschungsbericht

[20] MacDonald, Kevin: Navigating the waters of RFID Middleware. Dulles, Virginia, USA: ODIN technologies Laboratory, Juli 2005. - Forschungsbericht

[21] Holloway, Simon: Business Rules Management - Managing business rules of an organisation. London, England: Bloor Research, September 2009. – Forschungsbericht

[22] Gualtieri, Mike; Rymer, John R.; Heffner, Randy; Yu, Wallis: The Forrester Wave: Complex Event Processing (CEP) Platforms, Q3 2009 – for Application Development & Program Management Professionals – The Fledgling CEP Platform Market Is Vibrant, Competitive, And Dynamic. Cambridge, Massachusetts, USA: Forrester Research, August 2009. – Forschungsbericht

[23] Aberer, Karl; Hauswirth, Manfred, Salehi, Ali: Middleware support for the "Internet of Things". In: FGSN 2006 (Proceedings of the 5. GI-ITG KuVS Fachgespräch Drahtlose Sensornetze, Stuttgart, Deutschland, 2006-07-17/18). Stuttgart, Deutschland: Universität Stuttgart, 2006 (Technischer Bericht 2006/07). - Forschungsbericht

[24] Floerkemeier, Christian; Roduner, Christof; Lampe, Matthias: RFID Application Development With the Accada Middleware Platform. In: IEEE Systems Journal 1 (2007), Nr. 2, S. 82-94. - ISSN 1932-8184

[25] Soldatos, John: AspireRFID Can Lower Deployment Costs. In: RFID Journal. Hauppauge, New York, USA: RFID Journal, 16. März 2009.

[26] Swedberg, Claire: Pramari Launches Free Open-Source RFID Middleware. In: RFID Journal. Hauppauge, New York, USA: RFID Journal, 26. Oktober 2009.

[27] Binildas, C. A.: Service Oriented Java Business Integration - Enterprise Service Bus integration solutions for Java developers. Birmingham, England: Packt Publishing, Märtz 2008. – ISBN 978-1-847194-40-4

Deterministic technique for energy-efficient centralized clustering of wireless sensor networks

V. Delport, M. Gessner, T.D. Grossman, A. Singer
Chair of radio and communication technology,
Department of Electrical and Information Engineering, University of Applied Sciences Mittweida,
Technikumplatz 17, 09648 Mittweida, Germany
{volker.delport, gessner, grossman, asinger}@hs-mittweida.de

***Abstract*—This paper focuses on reducing power consumption in cluster-based wireless sensor networks. For this purpose we propose a deterministic, centralized, energy-efficient clustering technique, based on an adaptation of the hard c-means algorithm (HCM) with a pairwise nearest-neighbor initialization (PNNI). The numerical simulations show that the proposed clustering technique is suitable for increasing the lifetime of cluster-based wireless sensor networks in a rapid and deterministic manner.**

I. INTRODUCTION

A. Motivation

A wireless sensor network generally consists of a large number of small, low-power sensor nodes with wireless communication and limited computation capabilities. Sensor nodes usually operate with limited battery power. In most applications it may be inconvenient, and in some applications it is impossible to recharge the node batteries. In order to maximize the lifetime of the wireless sensor network, both the node hardware and the communication protocols must be designed to be energy-efficient.

In terms of the energy-efficient communication in wireless sensor networks, a cluster-based network organization is generally considered to be the most favorable approach (Fig. 1). In a cluster-based wireless sensor network, the sensor nodes are organized into clusters.

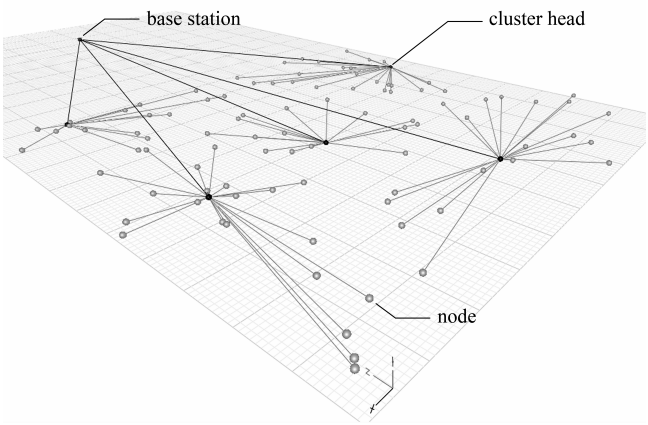


Figure 1. Cluster-based wireless sensor network

Each cluster has a cluster head, which is one of its sensor nodes. The cluster heads collect sensor data from their corresponding cluster nodes and transfer the aggregated data to a central sink or a base station, respectively. Obviously energy can be saved if an optimal clustering solution of the cluster-based wireless sensor network can be found. Furthermore, since the cluster heads dissipate much more energy than the ordinary sensor nodes, it makes sense to rotate the role of the cluster heads.

Most of the popular protocols for cluster-based wireless sensor networks like LEACH [1] have a significant disadvantage, because they utilize randomness in order to find the clustering solution. The LEACH-centralized protocol [1], and some of its successors, utilize simulated annealing for the selection of the cluster heads, but simulated annealing is based on randomness too. Simulated Annealing is able to approximate in probability the global optimum if two conditions are fulfilled. Firstly, a sufficient number of trials have to be carried out at a current temperature, so that the stochastic process receives a stationary distribution. Secondly, the temperature has to be slowly decreased from a sufficiently high value to a nearly zero value. Consequently simulated annealing guarantees statistically a nearly optimum solution, but it is quite time consuming as discussed by Delport [2].

Therefore, the motivation of this current research is to develop a clustering technique in order to maximize the lifetime of cluster-based wireless sensor networks in a deterministic and rapid manner

B. Basic approach

The first mentioned author has introduced several clustering algorithms for codebook design in vector quantization for the purpose of image data compression, e.g., in [2]-[5]. Vector quantization is a technique that has been investigated and used in speech and image coding for data compression. A vector quantizer maps a multi-dimensional vector space into a finite subset of reproduction vectors called a codebook. An input vector drawn from the vector space is approximated by its nearest code vector based on a certain distance measure, for example the Euclidian distance. The encoded output of a vector quantizer is the index of the representing code vector in the form of a binary word.

The process of codebook design in vector quantization is similar to the problem of the centralized clustering of a wireless sensor network. Therefore, our basic approach to solving the problem is the adaptation of relevant clustering algorithms for codebook design in vector quantization to the clustering of wireless sensor networks.

II. Energy Consumption Model

A. Energy consumption model of a wireless sensor node

A simple energy consumption model of wireless sensor nodes is used as per Heinzelman [1] and Furuta et al. [6]. In this model each sensor node is generally composed of five components: the battery, at least one sensor, the data processor, the transceiver electronics and the transmitter amplifier. The energy consumed per data bit by the transceiver electronics and by the data processor are assumed as constants E_{TxRx} and E_{DA}, respectively. The energy consumed per data bit by the transmitter amplifier E_{TxAmp} depends on the Euclidian distance $\|\boldsymbol{s}_{Tx} - \boldsymbol{s}_{Rx}\|$ between the transmitting and the receiving node. If this Euclidian distance is less than the predefined threshold d_0 a free space model with a path-loss exponent $\gamma = 2$ is assumed for the propagation channel, otherwise a multi-path model with a path-loss exponent $\gamma = 4$ is assumed.

B. Energy function of a data aggregation round

For the wireless sensor network we make some simplifying assumptions. All sensor nodes are immobile and able to reach both each other and the base station in a direct way, i.e., there is no multi-hop communication. The clustering of the sensor network is performed by the base station, which knows the position of all sensor nodes (centralized clustering). Based on the energy consumption model of a wireless sensor node described above, we assume the energy function (1) of a single data aggregation round in a cluster-based wireless sensor network.

$$E_{WSN} = \sum_{i \in S} b_i + \sum_{j \in H} b_j \tag{1a}$$

$$-L \sum_{i \in S} \sum_{j \in H} u\left(\mathbf{s}_i, \mathbf{h}_j\right) \cdot \left(E_{TxRx} + \left(E_{TxAmp,\gamma} \cdot \left\| \mathbf{s}_i - \mathbf{h}_j \right\|^{\gamma} \right) \right) \tag{1b}$$

$$-L \sum_{j \in H} E_{DA} - L \sum_{i \in S} \sum_{j \in H} u\left(\mathbf{s}_i, \mathbf{h}_j\right) \cdot \left(E_{TxRx} + E_{DA} \right) \tag{1c}$$

$$-L \sum_{j \in H} \left(E_{TxRx} + E_{TxAmp,\gamma} \cdot \left\| \mathbf{h}_j - \mathbf{s}_{BS} \right\|^{\gamma} \right) \tag{1d}$$

The set of ordinary sensor nodes is S. The set of cluster heads is H. In each data aggregation round the active sensor nodes and cluster heads have a certain positive battery level (1a). Each sensor node $\boldsymbol{s}$ sends L bit data to its assigned cluster head $\boldsymbol{h}$ (1b). The binary assignment variable $u(\boldsymbol{s}_i,\boldsymbol{h}_j)$ is equal to 1 if the sensor node $\boldsymbol{s}_i$ belongs to the cluster head $\boldsymbol{h}_j$, and otherwise $u(\boldsymbol{s}_i,\boldsymbol{h}_j)$ is equal to 0. The cluster heads receive the data from the corresponding sensor nodes and aggregate the received data with their own data to a data word of L bits (1c). Finally, the cluster heads send the aggregated data to the base station $\boldsymbol{s}_{BS}$ (1d).

III. Energy-Efficient hard c-means Algorithm

A. Introduction into the hard c-means algorithm

Since its introduction by Duda and Hart [7] the hard c-means algorithm (HCM) has been used as a clustering method in many application areas. In vector quantization the algorithm is better known as the generalized Lloyd algorithm (GLA) or the LBG algorithm since Linde, Buzo, and Gray introduced the idea of the hard c-means algorithm into codebook design [8].

Before the hard c-means algorithm starts the cluster number c is predefined and will be constant. By the so-called initialization process one representation point is chosen for each cluster as a temporary initial centroid by a more or less heuristic approach. In an association step each point in the considered environment is assigned to the nearest centroid. Afterwards the centroids are re-calculated using their corresponding points. As a result of the re-calculation step the centroids change their positions. The association and the re-calculation of the centroids are performed iteratively until the centroids do not move any more.

B. Energy-efficient hard c-means algorithm

Obviously the hard c-means algorithm would make an impact on the expressions of the energy function (1) where both the distances between the sensor nodes and the cluster heads (1b) and the distances between the cluster heads and the base station (1d) can be optimized. But, in order to use the hard c-means algorithm for an energy-efficient clustering of the wireless sensor network some further problems must be solved. It is very unlikely that the calculated cluster centroids are located in the same positions as the sensor nodes in the environment. Furthermore, a cluster head dissipates much more energy than an ordinary sensor node. Therefore, it makes sense to rotate the role of the cluster head between the sensor nodes by dynamical clustering. In order to solve these problems we propose the following adaptation of the hard c-means algorithm to an energy-efficient centralized clustering of wireless sensor networks.

Step 1: *Initialization:* Define the positions of the initial cluster centroids Z by using a suitable clustering initialization method.

Step 2: Set the energy weight $0 < \alpha \leq 1$, the termination threshold $\varepsilon > 0$, and the iteration index $r \leftarrow 1$, and calculate the weighted average battery level of the n active sensor nodes S with (2).

$$\overline{b_\alpha} = \frac{\alpha}{n} \sum_{i=1}^{n} b_i, \forall b_i > 0 \tag{2}$$

Step 3: Build c clusters S_j for which each sensor node $\boldsymbol{s}_i$ that satisfies (4) is assigned to its nearest cluster centroid $\boldsymbol{z}$ using (3)

$$\left(\mathbf{s}_i - \mathbf{z}_j\right)^2 = \min_{\mathbf{z}\in Z}\left(\mathbf{s}_i - \mathbf{z}\right)^2,\ i = 1\ldots n, j = 1\ldots c \qquad (3)$$

$$b_i \geq \overline{b_\alpha},\ \forall b_i > 0,\ i = 1\ldots n \quad , \qquad (4)$$

and calculate the average distortion error D_r with (5)

$$D_r = \frac{\sum_{i=1}^{n}\sum_{j=1}^{c} e(\mathbf{s}_i)\cdot u(\mathbf{s}_i)\cdot\left(\mathbf{s}_i - \mathbf{z}_j\right)^2}{\sum_{i=1}^{n} e(\mathbf{s}_i)} \quad , \qquad (5)$$

where $e(\boldsymbol{s}_i)$ and $u(\boldsymbol{s}_i)$ are defined as follows:

$$e(\mathbf{s}_i) = \begin{cases} 1 & if \quad b_i \geq \bar{b}_\alpha \\ 0 & if \quad b_i < \bar{b}_\alpha \end{cases} \qquad (6)$$

$$u(\mathbf{s}_i) = \begin{cases} 1 & if \quad \mathbf{s}_i \in S_j \\ 0 & if \quad \mathbf{s}_i \notin S_j \end{cases} \quad . \qquad (7)$$

Step 4: Calculate the cluster centroids $\boldsymbol{z}_j$ with (8)

$$\mathbf{z}_j = \frac{\mathbf{s}_{BS} + \sum_{i=1}^{n} e(\mathbf{s}_i)\cdot u(\mathbf{s}_i)\cdot\mathbf{s}_i}{1 + \sum_{i=1}^{n} e(\mathbf{s}_i)\cdot u(\mathbf{s}_i)},\ j = 1\ldots c \quad , \qquad (8)$$

where $\mathbf{s}_{BS}$ is the sensor node at the base station.

Step 5: Continue with $r \leftarrow r+1$ and step 3. If $\left(D_{r-1} - D_r\right)/D_r < \varepsilon$ skip step 4 and continue with step 6.

Step 6: *Cluster head selection*: Select c cluster heads $\boldsymbol{h}_j = \boldsymbol{s}_i$ by finding the sensor node $\boldsymbol{s}_i$ that satisfies (4) and (9)

$$\left(\mathbf{z}_j - \mathbf{s}_i\right)^2 = \min_{\mathbf{s}\in S}\left(\mathbf{z}_j - \mathbf{s}\right)^2,\ i = 1\ldots n, j = 1\ldots c \quad . \qquad (9)$$

Step 7: *Final clustering:* Build c clusters with (3), considering all active nodes and using the cluster heads $\boldsymbol{h}$ instead of the cluster centroids $\boldsymbol{z}$.

C. Comments

Please note that the energy-efficient adaptation of the hard c-means algorithm is due to the fact that the cluster building in step 3, the cluster centroids calculation in step 4, and the cluster head selection in step 6 consider just sensor nodes whose battery level is at least as much as the weighted average battery level of all sensor nodes that are alive.

For the initialization of the hard c-means algorithm in step 1 a pairwise nearest-neighbor initialization (PNNI) is highly recommended. The PNNI is a well-known method for codebook design in vector quantization [9]. The algorithm starts by considering each cluster point as its own cluster centroid, i.e., at the beginning, the clustering distortion error is zero. At each successive step, two clusters are merged into a new bigger cluster which is represented afterwards by the cluster centroid and the number of cluster members. The optimal cluster pair for merging at each step is the cluster pair which minimally increases the clustering distortion error after merging. The algorithm stops when the desired number of cluster centroids has been reached.

IV. EVALUATION

A. Simulation environment

In order to get a really valid impression how the proposed deterministic clustering technique works we have compared its performance with some of the results of Furuta et al. [6] where the clustering problem is regarded, among other things as an uncapacitated facility location problem (UFLP). For the UFLP based formulation of the energy function (1) the authors calculated the exact solution using the optimization software Xpress-MP (2005B).

Table I show the parameters which were used in our simulations as well as in Furuta et al. [6].

TABLE I. PARAMETERS USED IN THE SIMULATIONS

Parameter	Value
Number of sensor nodes	100
Cluster number c	5
Initial battery level per sensor node b_i	0.5 J
Energy consumption of the transceiver electronics E_{TxRx}	50 nJ/bit
Energy consumption of the data processor E_{DA} for data aggregation	5 nJ/bit
Energy consumption of the transmitter amplifier assuming the free space model $E_{TxAmp,2}$	10 pJ/bit/m^2
Energy consumption of the transmitter amplifier assuming the multi-path model $E_{TxAmp,4}$	0.0013 pJ/bit/m^4
Threshold of the distances between the transmitting and the receiving node for changing from the free space to the multi-path model d_0	87 m

As in Furuta et al. [6] a two-dimensional test field with 100 m x 100 m is chosen. Five different data sets, different from those used by Furuta et al. [6], with 100 randomly deployed sensor nodes each, were generated between the

position (x = 0, y = 0) and the position (x = 100, y = 100). The base station is located at the position (x = 50, y = 175). One of the used data sets is shown in Fig. 1.

For the simulations we use our own software tool Virtual Wireless Ad-hoc Networks (VIWIAN).

B. Results

As mentioned before, the purpose of our research is to maximize the lifetime of the sensor network. But, obviously the definition of the lifetime depends on the application service which is provided by the sensor network. In the current simulations we have concentrated on applications in which it is necessary that all sensor nodes stay alive as along as possible. Some examples for these application scenarios, where the network quality decreases considerably as soon as the first sensor node dies, could be housebreaking or fire detection.

In order to measure the lifetime we have used the survival rate which is defined as the percentage of active sensor nodes over all sensor nodes. Table II shows the survival rate versus the average (ave.) and the standard deviation (std.) of the number of the data aggregation rounds achieved by the proposed deterministic clustering technique, in comparison to the exact solution from Furuta et al. [6]. We have chosen the results from Furuta et al. [6] where the most data aggregation rounds were obtained for a survival rate of 99% with an energy weight of $\alpha = 0.9$. In comparison we obtained our best results for this survival rate with an energy weight of $\alpha = 1.0$.

Please note that the proposed technique needs on average only 5 ms per clustering and data aggregation round on a PC with an Intel® Pentium® 4 processor (2.4 GHz) and 1 GB RAM whereas Furuta et al. [6] reported needing about 5 s per round with nearly the same platform, in order to obtain the exact solutions.

TABLE II. THE SURVIVAL RATE VERSUS THE NUMBER OF DATA AGGREGATION ROUNDS

Survival Rate	**Exact solution from Furuta et al. [6] $\alpha = 0.9$**		**Energy-efficient hard c-means with PNNI, $\alpha = 1.0$, $\varepsilon = 0.01$**	
	Ave.	***Std.***	***Ave.***	***Std.***
99 %	912.8	13.50	917.6	16.88
90%	930.4	12.07	922.6	16.27
70%	943.8	11.21	930.4	14.84
50%	951.2	11.30	938.2	13.85
30%	957.4	10.71	943.6	12.16
10%	967.8	11.21	944.0	12.02

Ave.: Average, Std.: Standard deviation

V. CONCLUSIONS AND FUTURE WORK

A new deterministic, centralized clustering technique, based on an adaptation of the hard c-means algorithm with a pairwise nearest-neighbor initialization has been proposed for clustering of wireless sensor networks. It looks promising for use in real cluster-based wireless sensor networks.

The proposed clustering technique is currently implemented into a real test bed of about thirty hardware sensor nodes at our department. The sensor nodes, manufactured by Texas Instruments, are based on the system-on-chip solution CC2431 that combines an 8051 microcontroller unit (128 KB flash memory, 8 KB RAM) with an integrated 2.4 GHz IEEE 802.15.4 compliant radio transceiver CC2420. The radio transceiver is especially designed for low-power and low-voltage wireless applications [10]. The communication protocols, running on the sensor nodes over the IEEE 802.15.4 layers, have been developed and programmed by ourselves, without using an operation system like TinyOS or anything similar.

By means of this test bed we are looking to improve the energy consumption and radio propagation models in our simulation tool VIWIAN in order to approximate practical circumstances much better. Furthermore, we are in the process of extending models, simulations, and visualizations into the third dimension.

ACKNOWLEDGMENT

This research is supported by the Ministry for Science and Culture of the Free State of Saxony in Germany.

REFERENCES

[1] W. B. Heinzelman, Application-specific protocol architectures for wireless networks, Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, 2000

[2] V. Delport, Efficient Codebook Design in Vector Quantization by Parallel Simulated Annealing and Evolutionary, Proceedings of the IEEE Digital Signal Processing Workshop, Loen (Norway), pp. 219-222, 1996

[3] V. Delport, and M. Koschorreck, 'Genetic algorithm, for codebook design in vector quantisation', Electronics Letters, Vol. 31, No. 2, pp. 64-85, 1995

[4] V. Delport, Alternative Methods for Codebook Design in Vector Quantization, Proceedings of the IEEE Data Compression Conference, Snowbird (USA), IEEE Computer Society Press, pp. 485, 1995

[5] V. Delport, Beitrag zum optimalen Codebuchentwurf in der Vektorquantisierung, Dissertation, Tenea Verlag für Medien, Berlin, 2001

[6] T. Furuta, H. Miyazawa, F. Ishizaki, M. Sasaki, and A. Suzuki, A Heuristic Method for Clustering a Large-Scale Sensor Network, Proceedings of the Wireless Telecommunications Symposium, Pomona, USA, 26-28 April 2007

[7] R. Duda, P. Hart, Pattern Classification and Scene Analysis, Wiley, New York, 1973

[8] Y. Linde, A. Buzo, and R.M. Gray, An Algorithm for Vector Quantizer Design, IEEE Transactions on Communications, 28, 1, S.84-95, 1980

[9] W. H. Equitz, A New Vector Quantization Clustering Algorithm, IEEE Transactions on Acoustics, Speech, and Signal Processing, 37, 10, pp. 1568-1575, 1989

[10] Texas Instruments, http://www.ti.com/