Increasing Lifetime of Wireless Sensor Networks with Energy-Aware Role-Changing

Frank Reichenbach, Andreas Bobek, Philipp Hagen, and Dirk Timmermann

University of Rostock, Germany
Institute of Applied Microelectronics and Computer Engineering
Richard-Wagner-Street 31, 18119, Rostock-Warnemuende, Germany
{frank.reichenbach,andreas.bobek,philipp.hagen,dirk.timmermann}@uni-rostock.de

Abstract. Energy aware and robust self-organization is a challenging task in large, randomly deployed wireless sensor networks. In this paper, we achieve such a self-organization by introducing a hierarchical network structure and additionally roles that represent basic network functionalities like packet forwarding or data aggregation. These roles are exchanged between the participating nodes considering specific constraints. We are focusing on a long network lifetime, which strongly depends on the limited energy resources of each node. Therefore, the complex roles are released by nodes with critical battery levels and are assigned to nodes with more energy capacity left. With this approach, we achieve a uniform energy distribution over the whole network. Finally, we extend the overall lifetime of the network by 40% at continuous capability at all time. We demonstrate the proper function and the efficiency of the postulated protocol and we show its benefits by simulating an applicable "Forest Fire Scenario".

1 Introduction

Recently, tiny devices with sensors perform measurements of a physical phenomenon such as vibration, pressure, temperature, or humidity. The technical progress has been leading to smaller sensors, which are equipped with sensing hardware, their own processing units, memory, communication modules, and energy sources [1]. Hundreds or even thousands of these sensor nodes, are randomly deployed, self-organized, and communicate wireless. Normally, at least one special node, called the sink node, is installed at the edge of the sensor network. This node acts as the final collector of sensed data from the network's view, and provides gateway capabilities for further processing and monitoring.

When designing sensor networks a minimal power consumption is the key challenge. There are two main strategies to avoid too much energy consumption: Since sensors are spread out, it is expected that collected data is redundant when observing a common physical phenomenon under normal circumstances. Filtering and aggregating this data is a common technique to remove redundancy of measurements and therefore reduce the amount of data that has to be sent.

Secondly, instead of sending packets directly via long-range communication it is much more efficient to use short-range multi-hop routing [2]. Regarding the latter case, it is important to avoid node failures arising from battery exhaustion. Given these facts, a uniform energy distribution, considering the energy of each node, is strongly demanded. According to the network infrastructure, there are nodes with different power consuming functionalities assigned to them, such as aggregating, routing, or managing. These functions are further defined as *roles*.

This paper introduces *dynamic roles* to achieve efficient self-organization in sensor networks. Roles are changed dynamically within the network depending on the available energy of each node. We show that this approach results in a homogeneous power consumption of all nodes and consequently increases the overall network lifetime. Section 2 describes the advantages of self-organization and research in the field. Then, in Section 3 we describe our protocol to achieve dynamic role-change very efficiently. Comprehensive simulation results are shown in Section 4. Finally, we discuss these simulation results critically and conclude this paper in Section 5.

2 Self Organization in Sensor Networks

Self-organization in wireless sensor networks is essential to ensure proper measuring as well as communicating even if external influences of the environment (e.g. weather, impacts or electromagnetic fields) affect the network. Eventually, nodes join or leave the network, nodes are mobile, or the connection between nodes is asymmetric or interrupted. Because of the unknown position of each node in the beginning, the self-organization process implies multi-hopping by routing protocols and localization algorithms. Due to the following reasons, self-organization is substantially important:

- Initially, the topology of the deployed network is unknown and can change frequently.
- The participating number of nodes highly fluctuates due to node failures or battery exhaustion.
- Networks must be able to adapt to every kind of unknown environment and be aware of obstacles.

2.1 An Approach with Dynamic Roles

It is commonly accepted to define roles in a network that run on different nodes and thus must be exchangeable. For us, roles represent a specific algorithm that consumes a defined quantity of energy to be accomplished. Exemplarily, a parity check on physical layer will consume less energy than a complex encryption algorithm on an application layer. Nevertheless, these are two abstract roles that must be executed to fulfill the overall network function. It is conceivable to detach the roles of the specific hardware and to distribute them in the network depending on different parameters. In this paper, we will focus on low energy consumption on each node and in the end in the overall network. This approach enables:

- Fair distribution of all tasks in the network.
- A defined protocol by which nodes communicate with each other.
- The adaptation of tasks to specific parameters (e.g. a role release at minimal energy resources).
- Adding and removing roles, even if the sensor network is already deployed.
- The concentration of roles near a phenomenon (e.g. to strengthen a region or to minimize the communication overhead).

Different approaches already exist in literature, which we will be described in the following.

2.2 Related Work

Dynamic Clusterhead Gateway Switched Routing A well-known approach using roles is the "Dynamic Clusterhead Gateway Switched Routing" that presents a further stage of the "Dynamic Destination-Sequenced Distance-Vector Routing" (DSDV) by introducing hierarchical levels [3]. Thereby, efficient routing with small communication overhead is achieved. A hierarchy is established by so-called cluster-heads that manages the process within their specific clusters. Clusters are groups of nodes. Particularly, a cluster-head directly communicates with each node in the cluster. Clusters intersect topologically whereas a second role, the gateway, handles communication between them. Therefore, a gateway forwards packets from a cluster-head to the neighboring cluster-head. Extensive simulations showed that this approach is qualified to increase the overall network lifetime.

Low Energy Adaptive Clustering Hierarchy In the "Low Energy Adaptive Clustering Hierarchy" algorithm, data is collected from sensor nodes and then transmitted to a base station [4]. LEACH also considers a clustered network hierarchy where some of the nodes start executing the role of a cluster-head. They collect sensor data from all direct neighbors. This data is aggregated and then sent to a base station. Due to the high power consumption when transmitting data to a base station, the cluster-head role rotates within the network. The authors showed that this approach also leads to a longer lifetime of the overall network.

Connected Dominating Sets By extending the "Connected Dominating Sets" routing (CDS) from [5] with roles, a new approach is presented in [6] to achieve self-organization. New roles, the "Sensor Coordinator", the "Sensing Node", and the "Level 1 / Level 2 Backbone Nodes", are postulated in different levels of hierarchy. The so called "Sensing Zones" are created by the "Sensor Coordinator"-

nodes with “Sensing Nodes”. Thereby the authors defined a sensor dependent metric, called the “Cumulative Sensing Degree”. At present, dynamic role-change is not provided.

Generic Role Assignment In another approach, a role-based organization in a clustered network is extended by two new roles - the aggregator and the sink [7]. The first role aggregates information in the cluster to decrease the communication overhead caused by redundancy. The aggregator principally adopts the function of the cluster-head controlling the slaves that are neighboring nodes. The authors consider lists with all properties of the nodes. Additionally, tables with the description of each role and associated rules when assigning the roles are introduced. Finally, this achieves a generic role assignment.

We build up on some of the introduced concepts as well as dynamic roles to increase lifetime. Thereby, we focus on minimizing the power consumption on every node and consequently of the overall network. The following section describes our own protocol with all its concepts.

3 Concept of a Dynamic Role-Based Protocol

3.1 Architecture of the Network

To increase the lifetime of the sensor network by changing roles dynamically, the protocol has to fulfill some requirements. First the network must support self-organization of its nodes in an ad-hoc manner. Nodes can be added to or removed from the network transparently. It is noted that node mobility can also be interpreted as removing/adding nodes. We demand a mechanism for initializing and updating roles to react on changes of the environment and energy resources. For this purpose, nodes must be able to request services and information about their current state regarding resources of their neighboring nodes. Role assignment shall minimize global and local communication and lead to a uniform energy distribution over the whole sensor network.

We use clusters as the basis of our network architecture. A cluster is a logical structure that is formed by some nodes (see Fig. 1). It is managed by a special node called cluster-head. Communication between a cluster-head and its cluster-nodes is bidirectional, which means, they can communicate directly with each other. Each node belongs to a cluster and thus can reach at least one cluster-head by one hop or is a cluster-head itself. Here, a hop count defines the number of hops (relays or nodes) a message visits when traveling from source to destination. Nodes that reach more than one cluster-head are called gateways. They act as bridges between clusters and enable communication between cluster-heads. Sensing data is conveyed over gateways and cluster-heads to one or more sink nodes that serve as data collectors for further processing outside the sensor network.

The protocol differentiates between global and local communication. Marking global packets with sequence numbers facilitates reduction of global network traffic. Cluster-heads and gateways discard packets with same sequence numbers to avoid packet circulation. During role initialization, nodes analyze all received packets. After this, the communication is limited to the node's cluster. Nevertheless, it is still possible to send cluster broadcasts.

To reduce global traffic we introduce an aggregator. Such node collects data of sensor nodes within one cluster. In the case of nearly homogeneous data, the average value is determined and transmitted to the cluster-head. Moreover, we introduce a report cycle that defines the time between sending info packets. We want to emphasize that in contrast to other works, cluster-head and aggregator are assigned to different nodes. We expect a uniform distribution of energy in a cluster, since the cluster-head is already stressed by global communication, and therefore too many role changes are avoided.

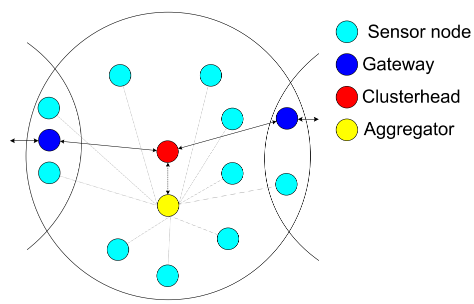

Fig. 1. Topology of 3 clusters and the participating roles

3.2 Basic Network Roles

The sensor role is the standard role for the nodes. After initialization and if no role was assigned, a node starts with the sensor role implicitly. Whenever a node has to release another role it falls back into the sensor role. Sensor nodes measure data and forward it to the aggregator periodically. The aggregator role is assigned to only one node per cluster. It collects data of all sensor nodes, analyzes them, filters out redundancy, and forwards it to the sink node via the cluster-head. The gateway role is responsible for inter-cluster communication. The cluster-head role has to forward global packets as well and manages the cluster and gateways to keep a consistent state.

Measuring and sending data within the network depends on a base cycle. This value is determined by the aggregator and is dynamically adapted to the current situation. The base cycle is identical to the time between two data packets sent by a sensor node. Currently, the sensor node measures data 10 times per base cycle. The averaged value is determined and sent to the aggregator.

If a measured value exceeds the last average values, packets are sent to the aggregator immediately. Since these packets present an exception, they are marked with a special exception flag. Moreover, all nodes in the cluster periodically send info packets to each other. Info packets contain among other information the energy level. All nodes except sensors need this information in order to assign a role to another node.

The main task of the aggregator is filtering and aggregating sensor data before forwarding it to the cluster-head. Furthermore, aggregators define the base cycle. If measured data is almost homogeneous over time, the base cycle can be increased. Cycles are implemented as levels. Higher levels consume more time than upper levels between two cycles. Instead of assigning an absolute time value, the aggregator assigns a level and increases (decreases) it.

3.3 Network Initialization

Self-organization of the network is separated into initialization phase and update phase. During initialization, a node executes an initializing algorithm, which is the same for each node (see Fig. 2). After that, it owns a role and starts the role specific task. This protocol uses delays to avoid network traffic storm in case of starting numerous nodes at the same time. After starting (booting) a node, it broadcasts an info packet. Info packets are sent periodically. They contain data about energy level, possible roles and current role. Information in these packets is extracted and stored in an info table that contains data of all neighbor nodes, their energy levels and currently assigned roles. If the node detects to be in range of a working cluster by finding cluster-head and aggregator, it joins this cluster and starts the sensor role. If a cluster-head but no aggregator is detected the node will wait for a role command that is sent by the cluster-head. In case of losing the connection to a cluster-head, or if no cluster-head is available at initialization, the node starts the cluster-head role and broadcasts a corresponding info packet. If no other nodes try to become a cluster-head after a random time, the new cluster-head sends a stand by message to consolidate the network and to start the cluster-head role. If more than one node falls into the cluster-head role during initialization, they all release their role and start the initialization routine once again.

We will describe in the following the process if a node has adopted the cluster-head role. All neighbor nodes are caused to check for gateway ability. This is the case if a node knows more than one cluster-head. Such nodes take over the gateway role and send an info packet to the cluster-head immediately. In the next step, the cluster-head chooses one gateway for each neighboring cluster according to their energy level and tries to set up a connection. If all connections fail (e.g. no neighbor cluster is accessible), the cluster-head gives up its role. If at least one connection was successfully established, it broadcasts a cluster-head initialization message. All nodes that are one hop away start participating in the cluster. Gateways that receive such message also participate in the cluster. The node with the highest energy capacity is assigned with the aggregator role. All other nodes receive a broadcast message to fall back into the sensor role.

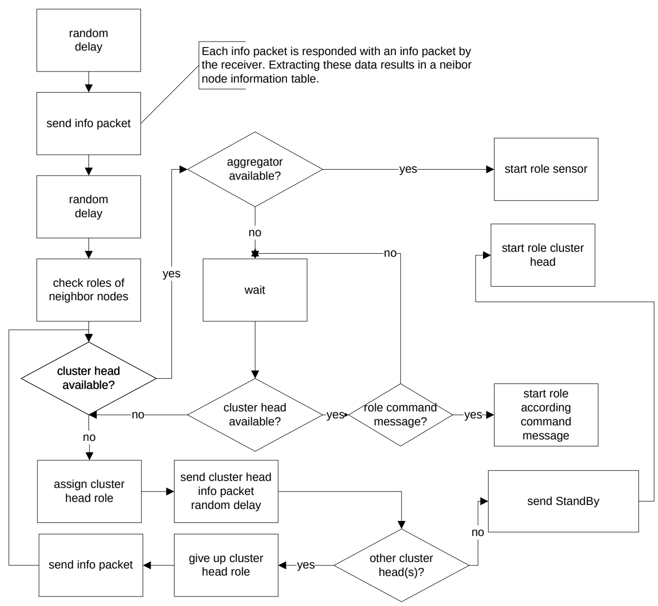

Fig. 2. Flow chart of the initialization phase

3.4 Dynamical Adaptation of the Roles

During runtime of the network, it is required to adapt the current energy situation by changing roles. Role changes can be enforced actively or when a missing role is detected. Aggregator and cluster-head control each other. This means, if one of them is broken, the other node chooses an appropriate node and initializes a role assignment. The same assignment is started if an aggregator or a cluster-head wants to leave its role.

When an aggregator releases its role actively, it sends a stand by message to the sensor nodes, which in turn, stops them to send data furthermore. After that, the aggregator's data base is transmitted in a two-step procedure. Each of these steps is acknowledged by the new aggregator node (Fig. 3). The data

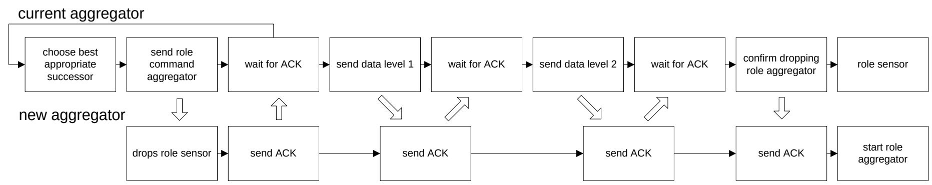

Fig. 3. Flow chart of a role reversal of the aggregator

base consists of mean, deviation, and exception values. The last acknowledge is sent to release the role at the old node and to start the role at the new node. Changing the cluster-head is very simple. The old cluster-head chooses a new node, sends a change message and waits for an acknowledge message. This is repeated until a new cluster-head is established.

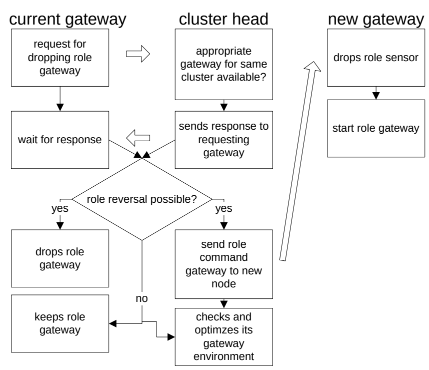

Fig. 4. Flow chart of the role reversal of gateway

If gateways want to release their role, they request the cluster-head for admission. The cluster-head evaluates possible nodes that reach the same neighbor cluster. If no successor is found, the gateway has to keep its role. Otherwise, the change can be applied to the old and the new node (Fig. 4).

Network Communication Communication is organized in logical layers. Each packet addresses a logical layer and implies one of the following semantics:

– Physical layer (0): all nodes that are one hop away analyze the packet
– Cluster layer (1): all nodes in the same logical hop composed of nodes within the same cluster like the sender analyze the packet
– Direct layer (2): packet is for a specific addressed node only
– Global layer (3): packet is for gateways and cluster-heads only

Three packet types for local communication are supported: sensor information message (SIM) used for info messages, sensor data message (SDM) and sensor command message (SCM). Furthermore, a global message (GM) packet is available to send sensor messages globally. Sensor messages are packed as payload in global messages (multiplexing and demultiplexing paradigm). To verify our concepts, we implemented the described concepts in the simulation tool J-Sim.

4 Simulation and Evaluation

Numerous simulation tools for wireless sensor networks are available. Well-known simulation platforms are NS2, OMNeT ++, J-Sim, SENSE, and Shawn [8–12]. We decided to use J-Sim as simulation framework. For our purposes, we needed to modify J-Sim by replacing some components to include our own energy model. This new model supports power consumption when processing data on a node, which was not possible yet. Further, we integrated hardware dependent power consumptions on bases of the MICA-2 system [13]. After these modifications, we were able to verify our concepts in extensive simulations.

This section presents simulation series designed firstly to make a proof of concept and secondly to validate and characterize the performance of a dynamic role-change-approach. In all simulations we used a sensor field with a size of 800×800 meters. The field was split into a grid of 10×10 subfields, each having a side length of 80 meters. All clusters are within one of these subfields. Between each pair of clusters the minimal distance is always the length of one subfield. Our goal by simulating was to be able to answer these questions:

– Do the initial assignment of roles and the dynamic change of roles work correctly?
– What are the implications of a specific energy capacity at which the role-change is initiated?
– What is the effect of a moving phenomenon in the sensor network? - simulating the “Forest Fire Scenario”

In all simulations, we started with a relative energy capacity of 100%, which is equal to an absolute capacity of 5000 mAh. Every activity (e.g. computation, transmission) on the node consumes a defined quantity of energy on basis of the power consumption of the Mica-2 motes postulated in [13].

4.1 Initial Role Assignment without Dynamic Role-Change

In this first simulation series, we studied the initial role-assignment after starting the network. We placed 6 clusters in the sensor field as illustrated in Fig. 5. Each cluster was directly connected with its neighboring clusters only. Cluster 6 forwarded data by a gateway to the sink of the network. We stimulated all sensor nodes with random generated sensor values that were out of a uniform distribution between 18 and 22. This is due to our intension later to use temperatures as sensor value. Moreover, to achieve clear results in an acceptable simulation time, we multiplied all calculated energy consumptions by factor 10, similarly. This did not impede the correctness of the results but affected the overall network energy to drop faster. We stopped the simulation after the last base cycle change was accomplished because at this point the initialization phase had been over.

Fig. 6 shows the energy capacity of each node over the time in cluster 4. Please note that in all following figures the time is in seconds multiplied by 10^3. First, the energy capacity of the aggregator will be exhausted, which can not be directly seen in the figure. Due to the constant sensor data, the aggregator raised its base cycle successively and thus the curve flattened more and more. Furthermore, the initial power consumption between 80 and 130 sec of the cluster-head and both gateways is marginal because the global communication had not been established. After 130 sec the global communication started, resulting in a fast drop of the curve. It can also be seen in Fig. 6 that all remaining nodes, excluding the special roles, took over the sensor node role and started the measuring process. After this phase, the power consumption of all nodes was almost linear.

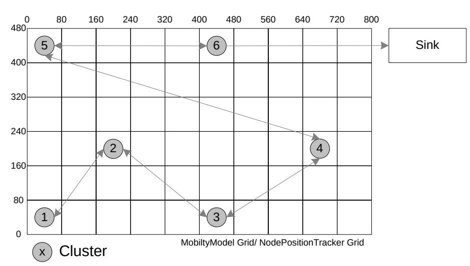

Fig. 5. Network configuration with 6 clusters and one data sink; network size is 800×800 meters (only 800×480 meters are shown); the communication between the clusters is marked with arrows

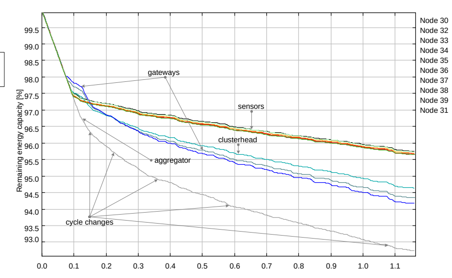

Fig. 6. Energy capacity [%] of each node over the time [sec] $\times 10^3$ in cluster 4 with 10 nodes

4.2 Initial Role Assignment and Dynamic Role-Change

In this simulation we explored the dynamic role-change after the initialization phase with varying energy differences. In order to do so, the energy difference indirectly specifies the moment of a role-change. In more detail, if the mean energy capacity of the whole cluster (E_{clmean}) subtracted by the current own energy capacity of a node (E_{own}) drops below the energy difference then the role must be released: $E_x = E_{clmean} - E_{own}$ (Switch if $E_x < E_{diff}$).

We considered networks with 10 and 20 nodes in each cluster. We justified the report cycle at constant 10 sec to achieve clear results regarding the role-change functionality. In addition, the power consumption was multiplied by 50 to minimize the simulation time. Following, we studied the implications in a cluster with 10 nodes only.

Implications of the Energy Difference Fig. 7 illustrates the result with an energy difference of 10%. All nodes switched the roles correctly. This ensured a more uniformly distributed power consumption over the sensor field. Moreover, no sensor node died prematurely and almost all nodes died in the same time interval of 600 sec. All curves fluctuated around an average straight line in a specific bandwidth. To analyze the influence of the energy difference, we run two more simulations with first a smaller value of 5% and secondly a higher value of 20%.

The results are presented in Fig. 8 and Fig. 9. The deviation at an energy difference of 5% was much smaller than the deviation at 20%. Besides, at 20% the lifetime of the cluster slightly increased. This was probably due to more communication overhead at role switching. In reality, nodes cannot predict the precise time at which the battery is exhausted due to electronic limitations of measuring a battery capacity. Thus, role-changes very close to the battery exhaustion are questionable.

In further simulations, we used the same test setup as before whereas 20 nodes formed a cluster (see Fig. 10-12). An energy difference of 10% was not sufficient to bother all nodes similarly. In contrast to the 10-node cluster, these results showed a longer lifetime at higher energy differences. The nodes with the important roles lived longer though the sensor nodes died successively above 2000 sec. Thus, the overall network functionality increased, but with fewer nodes that measured.

4.3 Dynamic Role-Change vs. the Static Case

To verify the benefits of dynamic roles, we compared simulation results of a dynamic network with a static network without role-changing. For this, we deactivated the dynamic role change in the following. Due to this modification, nodes with battery exhaustion die without transferring its role to another node. Thus, the remaining nodes have to detect role failure and have to reactivate the missing role. To achieve a quick detection of missing roles, we decreased the report cycle to 30 seconds. Besides, the configuration was similar to the previous ones. We simulated a cluster with 20 nodes.

Fig. 13 contains the characteristic curve of the energy capacity of a cluster, which shows a reduced lifetime by 600 seconds in comparison to the optimized dynamic network (see Fig. 12). The reason for this higher energy consumption

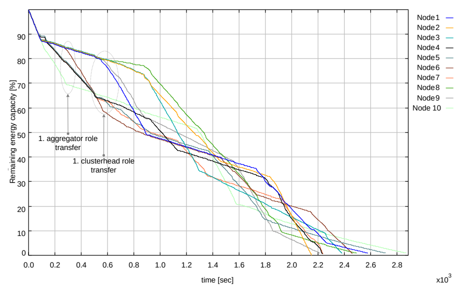

Fig. 7. 10 nodes in each cluster and an energy difference of 10%

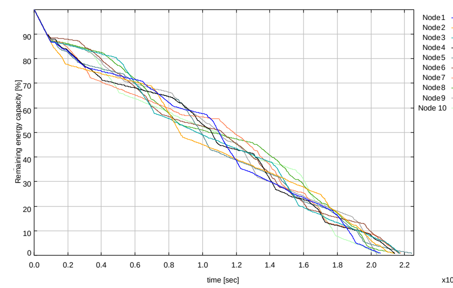

Fig. 8. 10 nodes in each cluster and an energy difference of 5%

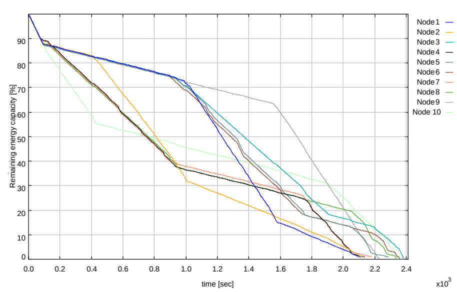

Fig. 9. 10 nodes in each cluster and an energy difference of 20%

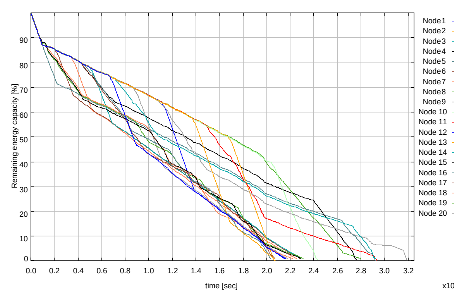

Fig. 10. 20 nodes in each cluster and an energy difference of 10%

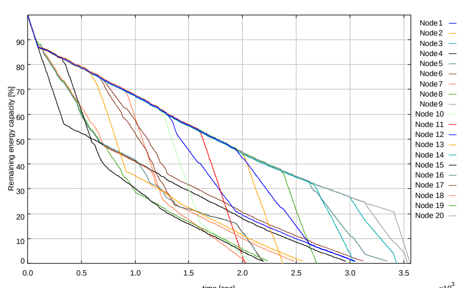

Fig. 11. 20 nodes in each cluster and an energy difference of 20%

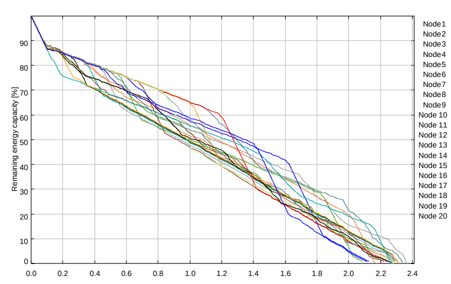

Fig. 12. 20 nodes in each cluster and an energy difference of 5%

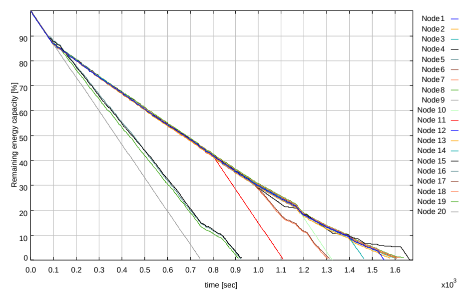

Fig. 13. Static network with initial role assignment only; the network consisted of 20 nodes in each cluster

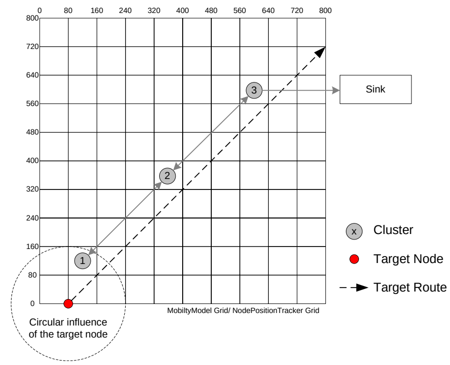

Fig. 14. Simulation of a moving sunbeam (the target node) that heated up the environment along a path with 3 clusters, each containing 10 nodes, and one data sink

was the higher frequency of info packets. The network did not work continuously with its full capability. In more detail, if one of the major roles died it needed between 30 an 45 seconds to got detected by the remaining nodes. In this detection period, no measured data left the cluster due to the missing aggregator. Moreover, missing cluster-heads or gateways implied problems in packet forwarding that reduced the performance of the cluster and the whole network. For example, alternative routing paths had to be explored. Hence, the efficiency of the global network organization decreased.

To summarize, in the static network the first node with the important aggregator role died at 750 sec followed by all other nodes until 1600 sec. Less than 80% of all nodes in the cluster were working after 60% of the network lifetime. However, in the dynamic network the first node died at 1500 sec. A dynamic cluster lived up to 40% longer than the static one. Moreover, the failure of the cluster-head at 900 seconds involved complex reorganization. Thus, the global consistency of the dynamic network was increased. Finally, the wasted energy of global routing and the reorganization had been reduced.

4.4 Simulation of the "Forest Fire Scenario"

In this last simulation, we verified our concepts in a conceivable application. In this scenario, the sun, represented by a sunbeam, heated up the surface along a path in the forest. Along that path, a forest fire could spark. We placed 3 clusters, each consisting of 10 nodes, along the critical path thus to be able to detect temperature changes. Cluster 3 was directly connected to the sink and each cluster was connected to its neighbor. The power consumption was multiplied by 10 and the test setup corresponded to the one in the sections before. The target node, representing the stimuli, started at 350 sec. The geometry of

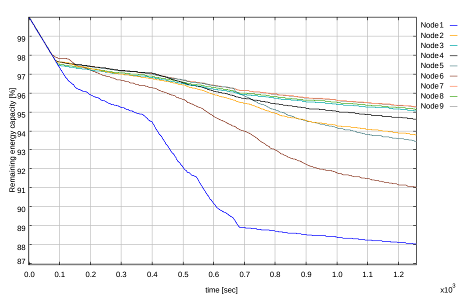

Fig. 15. Energy capacity of cluster 3 over the time in a network with a moving sunbeam

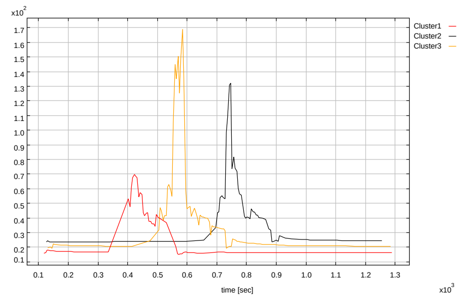

Fig. 16. Temperatures over the time that were measured at the data sink

this scenario is shown in Fig. 14. Beside the stimuli (temperature of the sun) of the target node itself, all clusters were stimulated by basic stimuli (temperatures of the environment) that ranged from 16 to 24.

The target node, representing the sunbeam, moved with a speed of 2.1 m/s. After simulation start the target node affected the sensor nodes in cluster 1. To reach the endpoint of the path the target node needed 500 sec where it lost at last his influence on cluster 3. The energy consumptions for cluster 1 is examplarily shown in Fig. 15. The moving target node effectuated exception packets initiated by the sensor nodes. Thus, the global communication increased. While the target node passed the clusters, the role of the aggregator as well as the role of the cluster-head changed.

The new aggregators retained in all three clusters its report cycle like the successor did. At the end position the target node lost its influence on the network and the energy consumption normalized in all clusters. The report cycle was reset to its maximum. The different maximums of the sensor values in Fig. 16 were due to the statistical distribution of the nodes in all 3 clusters. We finally verified the ability of adaptation of our protocol for the specific scenario.

5 Conclusion

This paper described a protocol that enables robust and efficient self-configuration that is strongly required in large and dynamic wireless sensor networks. For that, we introduced roles that represented the different fundamental network functionalities like communicating, forwarding, or precalculation. We demanded fair execution of these roles on all nodes similarly, independently of the different complexity each role possessed. By exchanging these roles between nodes, we achieved a uniform energy distribution over the whole network. This finally extended the lifetime of the network by 40%, compared to a static case where each node executes all tasks until its battery is exhausted. Additionally, we showed the proper function of the protocol in extensive simulations. We demonstrated the practical relevance of our concepts by simulating a real "Fire Forest Scenario" with a moving sunbeam along a path. In further research, we intend to study the implications of more roles (e.g. localization and encryption) and a network consisting of different hardware platforms.

Acknowledgment

This work was supported by the German Research Foundation under grant number TI254/15-1 and BI467/17-1 (keyword: Geosens).

References

1. I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, A Survey on Sensor Networks, *IEEE Communications Magazine,* August 2002, pp. 102-114.
2. J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, A Performance Comparison of Multi-Hop Wireless ad-hoc Network Routing Protocols, in *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking,* 1998, pp. 85-97.
3. P. E. Charles, and B. Pravin, Highly dynamic Destination-Sequenced Distance-Vector routing (DSDV) for mobile computers, in *Proceedings of the conference on Communications architectures, protocols and applications,* 1994, London, United Kingdom, pp. 234-244.
4. W. Heinzelmann, A. Chandrakasan, and H. Balakrishnan, Energyefficient Communication Protocols for Wireless Microsensor Networks, in *Proceedings of the 33rd Hawaii International Conference on System Sciences,* Volume 8, 2000, pp. 8020.
5. J. Wu, M. Gao, and I. Stojmenovic, On calculating power-aware connected dominating sets for efficient routing in ad hoc wireless networks, in *Proceedings of the International Conference on Parallel Processing,* 2001, pp. 346-356.
6. M. Kochhal, L. Schwiebert, and S. Gupta, Role-based hierarchical self organization for wireless ad hoc sensor networks, in *Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications,* 2003, pp. 98-107.
7. K. Roemer, C. Frank, P. J. Marrn, and C. Becker, Generic Role Assignment for Wireless Sensor Networks, in *Proceedings of the 11th ACM SIGOPS European Workshop,* 2004, pp. 7-12.
8. http://www.isi.edu/nsnam/ns
9. http://www.omnetpp.org
10. http://www.j-sim.org
11. http://www.cs.rpi.edu/ cheng3/sense
12. http://www.swarmnet.de/shawn/
13. V. Shnayder, M. Hempstead, R. Chen, G. W. Allen, and M. Welsh, Simulating the power consumption of large-scale sensor network applications, in *Proceedings of the 2nd international conference on Embedded networked sensor systems,* Baltimore, MD, USA, 2004, pp. 188-200.