



**HAL**  
open science

# Robustness and Performance Analysis of a Dynamic Sensor Network Scheduling Algorithm

Miklos Molnar, Gyula Simon, László Gönczy

► **To cite this version:**

Miklos Molnar, Gyula Simon, László Gönczy. Robustness and Performance Analysis of a Dynamic Sensor Network Scheduling Algorithm. FTDA-DN: Future Trends on Design and Analysis of Dynamic Networks, Jul 2008, Hong-Kong, China. lirmm-01680400

**HAL Id: lirmm-01680400**

**<https://hal-lirmm.ccsd.cnrs.fr/lirmm-01680400>**

Submitted on 10 Jan 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Robustness and Performance Analysis of a Dynamic Sensor Network Scheduling Algorithm

Miklós Molnár  
IRISA/University of Rennes 1  
Campus Universitaire de  
Beaulieu  
35042 Rennes Cedex, France  
molnar@irisa.fr

Gyula Simon  
University of Pannonia  
Veszprém, Hungary  
simon@dcs.vein.hu

László Gönczy  
Budapest University of  
Technology and Economics  
Budapest, Hungary  
gonczy@mit.bme.hu

## ABSTRACT

This paper presents the robustness and performance analysis of the Controlled Greedy Sleep algorithm, which was designed to provide  $k$ -coverage in wireless sensor networks. The aim of this algorithm is to prolong network lifetime while ensuring QoS requirements in a dynamic manner. We investigated how the network can be strengthened to improve performance characteristics, and how this algorithm ensures graceful degradation (i.e., how the network will provide less accurate measurement data as sensors become unavailable). We also test the robustness of the algorithm by measuring the effect of message loss due to communication errors. We compare the results to those of a very known and frequently used random algorithm. Our performance tests are based on simulations results.

## Categories and Subject Descriptors

C.2.1 [Computer-Communication Networks]: Network Architecture and Design—*Wireless communication*; C.2.3 [Computer-Communication Networks]: Network Operations—*Network management*; C.4 [Performance of Systems]: Reliability, availability, and serviceability

## General Terms

Algorithms, Measurement, Performance

## Keywords

sensor networks, energy conservation, coverage

## 1. INTRODUCTION

Wireless Sensor Networks (WSNs) are among the most current research topics in the field of distributed systems and ad-hoc networks. As the price of sensors is becoming lower and lower, these kind of devices are used in several application environments. However, the energy consumption

of a WSN is a critical issue to be solved, since individual sensors have limited power capacity and the entire network still has to meet relatively strong lifetime requirements. Therefore, *scheduled sleeping* is a key functionality which has to be provided in a WSN infrastructure. Usually, sensors either *sleep* or are *awake* for a given  $T$  time of period, thus saving energy. Random and coordinated sleeping algorithms were proposed to organize the scheduling of a WSN. On the other hand, the measurement objective and the accuracy of sensor measurements can implicate a  $k$ -coverage criterion. This means that every point has to be monitored at least by  $k$  sensors to get proper measurement results. As the batteries of sensors deplete, the  $k$ -coverage cannot be guaranteed longer than a certain time threshold, which is called  $k$ -lifetime of the network. Users, however, may also be interested in receiving less accurate data, i.e., a WSN should also provide a degraded service as long as possible.

This paper evaluates performance and robustness of the Controlled Greedy Sleep (CGS, [15]) algorithm which aims to provide a trade-off between network lifetime and measurement accuracy so that  $k$ -coverage is guaranteed as long as the network topology enables it. CGS is a novel solution considering both the remaining energy of sensors and the importance of their measurement results. The algorithm tolerates the loss of messages (cf. [15]), which may cause only over-coverage, i.e., sensors may stay unnecessary awake.

Investigating the performance and robustness of CGS algorithm, we wanted to see how it compares to other existing solutions according to the following criteria:

- Effect of network topology on the algorithm. We analyze cases where sensors are deployed in a grid topology, or by a uniform random distribution process. These topologies are the most frequent and easiest to analyze. We also present how strengthening the "edges" or "corners" of the topology effects the behavior of a WSN.
- Spatial distribution of  $k$ -lifetime for different values of  $k$ . This corresponds to the degradation of the service provided by the WSN.
- Number of awake sensors and provided QoS as a function of communication message loss. This shows the *robustness* of the algorithm.

The rest of this paper is organized as follows: Sec. 2 presents related papers and Sec. 3 reviews the algorithm. Sec. 4 presents the analysis results in details. Sec. 5 concludes the paper and discusses further research issues.

...

## 2. RELATED WORK

Several challenges exist by employing a Wireless Sensor Network (WSN). Dependability in a dynamic utilization context should be analysed from the point of view of data measurement and data transmission, which are major functionalities of a WSN [1]. To ensure fault tolerance at different architectural levels, different solutions are proposed [7]. In a general WSN architecture several sensor nodes (SN) and at least one but eventually more Base Stations (BS) or sinks are considered. While a BS can be powered in a wired manner in some cases, SNs are generally battery powered [14], therefore the power consumption of battery dependent elements determines the lifetime of the network.

Sensors are generally deployed with high redundancy in the target area. For a given application each point of the area must be monitored by at least  $k$  SNs. Basically, this  $k$ -coverage requirement is associated with the measurement functionality [9] but it also has trivial implications to connectivity issues [2, 17] as well.

Several propositions exist to prolong the network lifetime by turning off a subset of available sensors for a time interval and scheduling this sleep interval so that awake sensors still provide continuous service (cf. in [10, 4, 17]). Lifetime longevity and the network operability require efficient dynamic scheduling algorithms. Mainly, the scheduling procedures can be classified into two representative groups: random and coordinated scheduling algorithms [13].

A distributed, random sleeping algorithm can be found in [5] where nodes make local decisions on whether to sleep or to join a forwarding backbone. The decision of each node is based on an estimation of the number of the neighboring sensors. In [11] the authors propose a simple randomized scheduling for dense and mostly sleeping sensor networks. They suppose that there are many redundant sensors and one can compute an identical duty cycle for individual sensors. In the proposed Randomized Independent Sleeping algorithm, time is divided into periods. At the beginning of each period, each sensor decides whether to go to sleep (with probability  $p$  computed from the duty cycle) or not, thus the lifetime of the network is increased by a factor close to  $1/p$ . This solution is very simple and does not require communication between sensors, but there is no guarantee for coverage nor for network connectivity. Furthermore, this solution cannot adapt to inhomogeneous or mobile sensor setups.

The authors in [13] propose a Role-Alternating, Coverage-preserving, Coordinated Sleep Algorithm (RACP). Each sensor sends a message periodically to its neighbourhood containing state information. An explicit acknowledgment-based election algorithm permits to decide the sleep/awake status. The coordinated sleep is more robust and reduces the duty cycle of sensors compared to the random sleep algorithm. The topology can affect the behaviour; thus the sensors can adapt their sleeping to the needs. The price of the performance is the significant communication overhead increasing power consumption. In [12] the asymptotic behavior of coverage in large-scale sensor networks is studied. For the  $k$ -coverage problem, formulated as a decision problem, polynomial-time algorithms (in terms of the number of sensors) are presented in [9]. A comprehensive study on both coverage and connectivity issues can be found in [8].

Performance evaluation of the WSN architectures and network protocols helps to choose the right WSN [3]. In [6] a Markov model based performance analysis of WSN is de-

scribed. The authors investigate the system performance in terms of energy consumption, network throughput and data transmission delay.

To make a good trade-off between  $k$ -coverage guarantee and network resource utilization, we proposed in [15] a new coordinated algorithm. This proposition takes into account both the power status and the sensing assignment of the SNs. Based on this information, a so-called *drowsiness factor* is computed periodically for each sensor. Then this factor, reflecting the status of the sensors, is used as priority to decide which sensors need power conservation the most and thus can be sent to sleep. A distributed election algorithm was also proposed which uses only locally available information to decrease the communication overhead. The Controlled Greedy Sleep Algorithm requires only a few messages to be broadcasted locally from every node to the direct neighbors in each period.

## 3. THE ANALYSIS PROBLEM

Before the description of the analyzed algorithm, we provide a quick overview on the dynamic  $k$ -coverage scheduling problem.

### 3.1 Network Model and Objectives

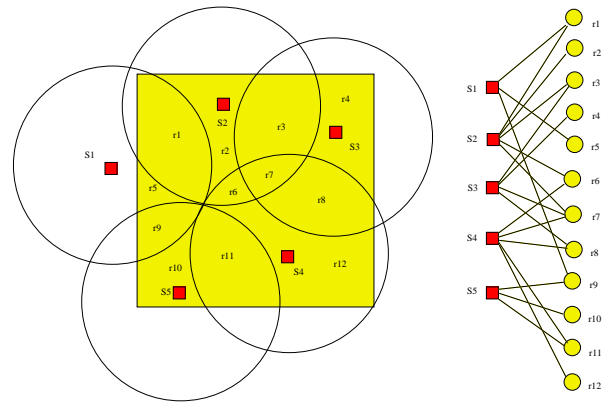


Figure 1: Relations between sensors and regions

The different SNs participate to measure and survey a part of the target area. To simplify, we suppose that the part of target that is measured by a SN corresponds to a disk. Several SNs control a same set of points corresponding to the intersection of the sensing disks, which we will call region (see Fig. 1). Let  $S$  and  $R$  be the set of SNs and the set of regions respectively. The sensing assignment between them can be represented by a bipartite graph  $G(S \cup R, E)$ . In  $G$  there is an edge between region  $r$  and sensor  $s$  if and only if  $s$  covers the region  $r$ . The simple, static  $k$ -coverage problem is to find a sub-graph  $G'(S' \cup R, E')$  with a sub-set  $S' \subset S$  so that for all vertices  $r \in R$  the degree is at least  $k$ . The *minimal  $k$ -coverage problem* is to find a sub-graph that solves the  $k$ -coverage problem with minimal number  $|S'|$  of sensors. That is a particular case of the Minimal Cover Vertex Set problem, which is a NP-complete optimization problem.

The goal of the *dynamic scheduling algorithms* in WSN is more complex than the computation of static minimal  $k$ -covering sets. These algorithms aim to prolong the lifetime

of the sensor network with the help of the alternation of appropriate k-covering sets (and so with the help of the alternation of awake and sleep states of SN) minimizing the power consumption. Let  $c_s$  be the energy consumption of a SN per time unit (to model the problem, we suppose that all SNs use the same power when they are in awake state). Let  $S_k$  the set of the sub-sets of  $S$  which solve the static k-coverage of the target area. The goal of the scheduling algorithm is to find the ordered set  $OS^*$  of sub-sets  $S^* \in S_k$  which maximize the lifetime of the WSN: passing from one sub-set  $S^* \in OS^*$  containing alive sensors to the following one by respecting the order of the scheduling, such that the duration time of  $OS^*$  is maximal.

To keep tabs on the power level of SNs and on the k-coverage criteria in parallel, a new metric describing the state of SN was proposed recently in [15]. The proposed drowsiness factor models the priority of SNs to go to sleep and is computed at each period for each node. Supposing that the node  $s$  has remaining energy  $E_s$ , its drowsiness factor is defined as follows:

$$D_s = \begin{cases} \frac{1}{E_s^\alpha} \sum_{r \in R} \Phi_r & \text{if } \Phi_r > 0, \forall r \\ -1 & \text{otherwise.} \end{cases}$$

where  $\alpha$  is a positive constant (e.g.  $\alpha = 2$ ), and  $\Phi_r$  is the coverage ratio of region  $r$  defined as follows:

$$\Phi_r = \begin{cases} \frac{1}{C_r - k} & \text{if } C_r > k \\ -1 & \text{otherwise.} \end{cases}$$

Here  $C_r$  is the degree of region  $r$  in  $G$ . The coverage ratio  $\Phi_r$  is positive if the region is over-covered, i.e. more than  $k$  sensors could cover region  $r$ . It is negative if region  $r$  is not over-covered: in this case all SNs possibly covering  $r$  must be in awake state. Moreover, the smaller the energy of a SN, the larger its drowsiness. A sensor participating in many regions that have low over-coverage is likely to participate in more possible solutions than sensors covering regions also covered by many other sensors. Thus a heuristic property is included in  $D_s$  to increase the lifetime of the network: sensors participating in regions only slightly over-covered have larger drowsiness. The drowsiness factor for each sensor includes the sum of the coverage ratios of the regions the sensor is able to observe. This property enforces the sensors in critical positions to go to sleep whenever it is possible, to conserve their energy for times when their participation will become inevitable. Negative drowsiness indicates that the SN is not allowed to sleep. To realize the proposed algorithm and in order to organize communication between SN, the communication radius is supposed at least twice of the sensing radius. This assumption automatically provides network-wide communication if 1-coverage in sensing is provided [17]. With the help of this assumption, it is easy to see those sensors participating in the coverage of at least one region covered by a sensor  $q$ , are inside the communication radius of  $q$ . The sensors know their own coordinates and the observed area. Thus SNs handle only regions that are inside of the target area.

### 3.2 The distributed CGS algorithm

In the proposed solution nodes have local information on their neighborhood only. Thus instead of  $G$  each sensor node  $q$  will use a locally known sub-graph  $G_q(S_q \cup R_q, E_q)$ . This sub-graph contains geographical regions  $R_q$  covered by  $q$ , the set  $S_q$  of sensors which participate the coverage of at

least one region of  $R_q$ , and edges between them.

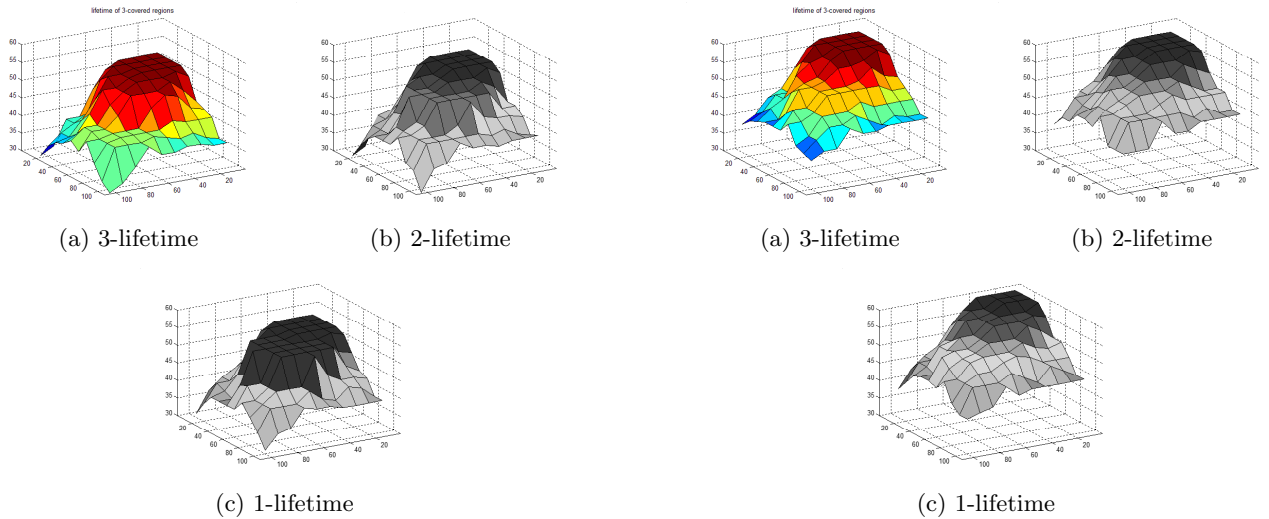
The scheduling aims to preserve the power of sensors in critical state with the help of sleep periods. A sensor  $s$  can go to sleep if its neighbors with larger drowsiness factor decided their state for the next period and  $s$  has no critical (not over-covered) region to monitor. For this, each sensor should know the drowsiness factor of the neighbors and the decision of neighbors with larger factor. To minimize the local communication, a communication delay (DTD) is associated with each sensor. This delay is inversely proportional with the drowsiness factor. So the sensors with large factor broadcast their decision earlier. Only the awake state decision should be broadcasted explicitly, in this way the communication overhead can be minimal.

The main steps of the Controlled Greedy Sleep (CGS) Algorithm are described in the following:

1. Wake up all sensors at the beginning of the period.
2. Alive nodes broadcast local *Hello* messages containing their node locations. Based on received *Hello* messages each node  $s$  builds up its local set of alive neighbor nodes  $S_s$  and generates the local bi-partite graph  $G_s(S_s \cup R_s, E_s)$ , and then it calculates its drowsiness factor  $D_s$ .
3. Based on  $D_s$  each node  $s$  selects a Decision Time Delay ( $DTD_s$ ). Small drowsiness means large DTD, large drowsiness means small DTD. These delays provide priorities when nodes announce their Awake Messages (AM). Each node  $s$  broadcasts its  $DTD_s$  and starts collecting other nodes' DTD and AM messages. From the received DTD messages each node builds a Delay List ( $DL_s$ ), and from the received AM it builds a List of Awake Neighbors ( $LAN_s$ ).
4. After  $DTD_s$  time elapsed the node  $s$  makes a decision based upon  $LAN_s$  and  $DL_s$ :
  - if all regions in  $R_s$  can be k-covered by only nodes present in  $LAN_s$  and/or nodes  $q$  present in  $DL_s$  for which  $DTD_q > DTD_s$  then go to *sleep*
  - otherwise stay *awake* and broadcast an AM to inform other nodes that node  $s$  will stay *awake*.

In step 4 nodes go to sleep in a greedy manner: if the coverage problem can be solved with the already known awake neighbors and other nodes with lower drowsiness then the node greedily elects to sleep and leaves the problem to those already being awake and those who haven't decided their status yet.

Finally, let us notice that the communication overhead of the algorithm depends on the length of periods and can be low. In each period every node broadcasts only at most three messages (two if it the node will go to sleep, three otherwise). In addition to this, nodes must stay awake in order to complete the election process. During this extra  $T_e$  time nodes consume energy. The communication and awake-time overhead can be neglected if the length  $T$  of a period is significantly longer than  $T_e$ , which is true in most practical cases. The determination of the optimal length  $T$  is out of scope of our present paper.



**Figure 2: k-lifetime of CGS in the random deployment experiment**

**Figure 3: k-lifetime of CGS in the random topology with extra nodes on the perimeter**

#### 4. ROBUSTNESS AND PERFORMANCE ANALYSIS

The CGS algorithm and the random  $k$ -coverage algorithm were simulated in Prowler, a probabilistic sensor network simulator [16]. The simulator parameters were set to model the Berkeley MICA motes' MAC layer. The radio propagation model includes realistic effects, e.g. fading, collisions and lost messages.

The main requirement against the algorithms were to provide reliable measurement data, i.e.,  $k$ -coverage for all regions. We expected that due to its probabilistic nature the random algorithm will not be able to guarantee  $k$ -coverage. On the other hand, we expected that the behavior of the CGS algorithm will show a "graceful degradation", i.e., coverage criteria would be met for different  $k$  values in a similar way, therefore applications can count on less accurate observations in a schedulable manner. We also expected that since SN with less neighbors will be able to go sleep only a few times, coverage will be first degraded at the edges of the network.

Two sensor deployment topologies were analyzed by simulation. (1) The first case contains 10x10 sensors in a grid topology. Assumed sensing radius is 15m and nodes have an initial energy of 20 unit where one period consumes 1 unit. To enhance lifetime in critical parts of the network, we used an extended grid topology where "corners" are strengthened with extra sensors. Grid topology is easy to analyze, however, its application in real life measurement environment can be problematic due to strict deployment requirements. (2) In many applications the observed area is not easily accessible, thus random placement methods should be used (e.g. deployment from UAV). To generate random sensor deployment, we used a topology with 100 nodes and a sensing radius of 30m. Initial node energy is considered to be 10 while one active period consumes 1 unit of energy. Often the perimeter can be accessed more easily thus the placement of extra nodes to the right position is doable. Therefore, the extended topology corresponds to the random deployment

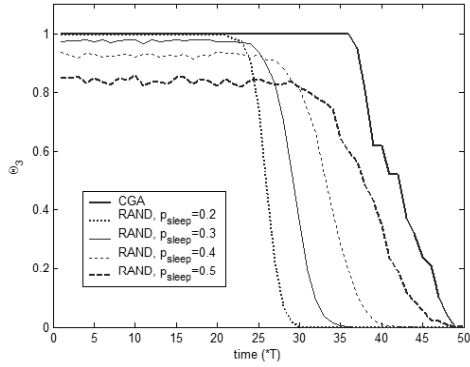
with extra sensors in the perimeter.

First we wanted to see how the QoS (in terms of measurement accuracy) provided by the network degrades. Therefore, we performed simulation to see how the randomly deployed network compares to that with additional nodes. Results (guaranteed  $k$ -lifetime of regions) of CGS are shown in Fig. 2 and Fig. 3 as a spatial distribution.

The  $k$ -lifetime values of each 1mx1m region were measured. The goal was to provide 3-coverage, but the 1 and 2-lifetimes were also measured for each region. The lifetime of the central regions was significantly longer and the network started to degrade at the lower-covered perimeters, especially at the corners. As expected, the algorithm provides longer  $k$ -lifetime for smaller  $k$  values. Including 12 extra perimeter nodes (i.e. the number of nodes increased by 12% ) has dramatically changed the performance of the network. The 3-lifetime of the network increased from 29 to 38 periods (increase of 31% ). The central regions were still covered for a longer time but the difference between the first and last uncovered region also decreased (from 23 to 20 periods) and the distribution of the used energy became more balanced.

Fig. 4 shows results of experiments in the extended grid topology. The measurement objective was to estimate how the percentage of  $k$ -coverage for the entire network changes in time. The random algorithm was simulated with four different  $p$  probabilities for sleeping decision. As it is shown, only the CGS guarantees the  $k$ -coverage. Although random algorithm with low  $p$  value provide high coverage ratio (a value close to one), the price of this is the relatively low network lifetime. CGS provides  $k$ -coverage for 38 periods, while for the random algorithm the QoS starts to decrease very fast after 25-30 periods (from a lower initial value), for all  $p$  parameters.

As WSNs usually operate in a very noisy environment, we wanted to test how the CGS tolerates communication message losses. Moreover, communication congestions may involve messages losses. When two neighbor sensors having the same DTD value broadcast their decision simulta-



**Figure 4: Degradation of service in time, using the extended grid topology for the CGS and random algorithm**

neously, these messages may be lost. Although lost messages may cause "over-coverage", i.e. sensors remain awake unnecessarily, the algorithm is tolerant against communication faults and the QoS is provided, as long as there are enough available sensors. We used the random topology for these simulations. As Fig. 5(a) shows, message loss does not have a major effect on the guaranteed k-lifetime of CGS, but speeds up the degradation of the network (ratio of k-covered area). Since the random algorithm does not use communication, we did not consider message loss in this case. Instead, we used different sleeping probabilities, as shown in Fig. 5(b). The results are quite similar to those shown in Fig. 4.

Finally, Fig. 6 shows the number of awake sensors in time. Although the QoS of CGS is resilient against message loss, the number of awake sensors increases as the probability of message loss increases. As a logical consequence, sensors lose their energy sooner and deplete. This shows that an attack or artificial "noise" in the environment can decrease the network lifetime but does not really effect measurement accuracy which can be an important aspect in critical applications.

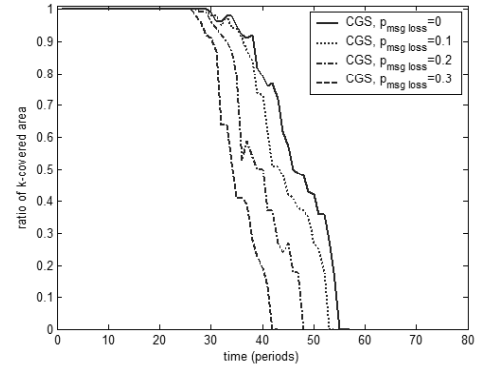
For the random algorithm, at the beginning the number of awake sensors corresponds to  $1-p$  and then sensors disappear very fast, which is in fact the reason for the unreliable network service.

## 5. CONCLUSIONS AND FUTURE WORK

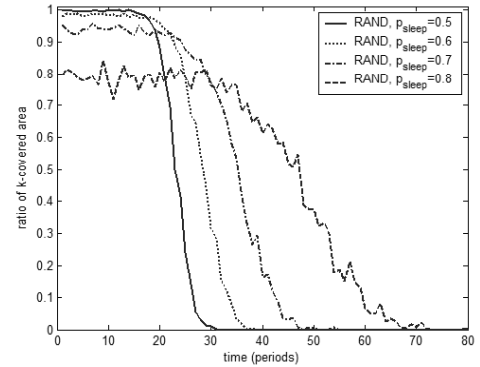
We analyzed two dynamic scheduling algorithms for WSNs, providing accurate measurement data and prolonging network lifetime. During the analysis we used the Prowler probabilistic network simulator.

Our measurement results showed that the CGS algorithm outperforms the random one both in terms of guaranteed QoS (guaranteed level of k-coverage) and network lifetime. Robustness of CGS was also tested against message loss. Importantly, CGS provides graceful degradation, i.e., the measurement accuracy decreases with a lower speed as a function of time. This is a very important practical aspect, since a lot of applications can still process measurement data of a lower quality.

We also experienced that during deployment, it is worth



(a) QoS characteristics of the CGS algorithm as a function of message loss



(b) QoS characteristics of the random algorithm as a function of sleeping probability

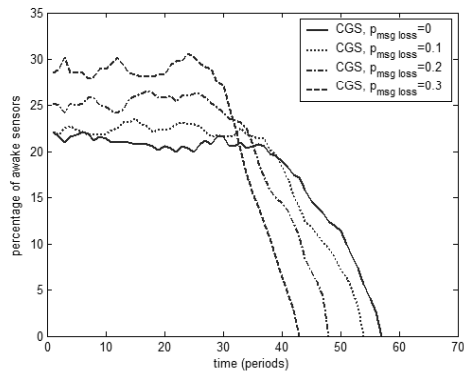
**Figure 5: Degradation of QoS in the random topology**

placing more sensors to the corners and edges of the network since expected k-lifetime increases significantly with a relatively low cost overhead (caused by the additional sensors).

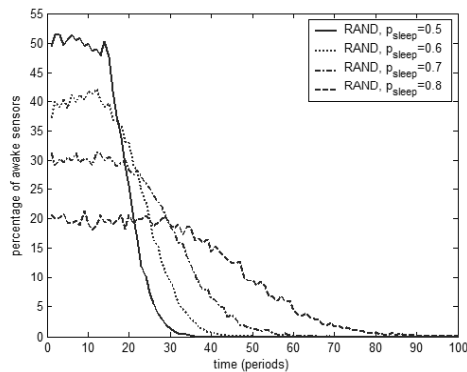
A further research questions to investigate is the effect of period time on the system lifetime. As the communication overhead is supposed to be relatively low, we expect that unless we choose very small periods, CGS will not be effected by this. Period time is more likely to effect random algorithms, and those with more communication needs.

As a WSN may provide multiple services, an interesting question to answer is how to manage multifunctional sensors, i.e., a heterogeneous sensor environment where the measurement capabilities of nodes also has to be considered at scheduling.

Finally, we plan to investigate higher level QoS characteristics as a function of applied WSN strategy and topology. We will also perform a security test of CGS to see how resilient it is against malicious messages and how security of the algorithm can be augmented.



(a) CGS algorithm with different message loss probabilities



(b) Random algorithm with different sleeping probabilities

**Figure 6: Number of awake sensors in the random topology**

## Acknowledgements

This work was partly supported by the "Dependable, intelligent networks" (F- 30/05) project of the Hungarian-French Intergovernmental Cooperation Program and by the Hungarian Government under contract NKFP2-00018/2005.

## 6. REFERENCES

- [1] N. Ahmed, S. S. Kanhere, and S. Jha. The holes problem in wireless sensor networks: a survey. *SIGMOBILE Mob. Comput. Commun. Rev.*, 9(2):4–18, 2005.
- [2] H. M. Ammari and S. K. Das. On computing conditional fault-tolerance measures for k-covered wireless sensor networks. In *MSWiM '06: Proceedings of the 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems*, pages 309–316, New York, NY, USA, 2006. ACM Press.
- [3] G. Anastasi, A. Falchi, A. Passarella, M. Conti, and E. Gregori. Performance measurements of motes sensor networks. In *MSWiM '04: Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems*, pages 174–181, New York, NY, USA, 2004. ACM Press.
- [4] L. S. Brakmo, D. A. Wallach, and M. A. Viredaz. Sleep: a technique for reducing energy consumption in handheld devices. In *MobiSys '04: Proceedings of the 2nd international conference on Mobile systems, applications, and services*, pages 12–22, New York, NY, USA, 2004. ACM Press.
- [5] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris. Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. *Wirel. Netw.*, 8(5):481–494, 2002.
- [6] C.-F. Chiasserini and M. Garetto. Modeling the performance of wireless sensor networks. In *INFOCOM*, 2004.
- [7] A. S.-V. Farinaz Koushanfar, Miodrag Potkonjak. Fault tolerance in wireless ad-hoc sensor networks. In *Proceedings of IEEE Sensors 2002*, June 2002.
- [8] A. Ghosh and S. K. Das. Coverage and connectivity issues in wireless sensor networks. pages 221–256. John Wiley, 2006.
- [9] C. Huang and Y. Tseng. The coverage problem in a wireless sensor network. In *WSNA '03: Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications*, pages 115–121, New York, NY, USA, 2003. ACM Press.
- [10] R. Krashinsky and H. Balakrishnan. Minimizing energy for wireless web access with bounded slowdown. In *MobiCom '02: Proceedings of the 8th annual international conference on Mobile computing and networking*, pages 119–130, New York, NY, USA, 2002. ACM Press.
- [11] S. Kumar, T. H. Lai, and J. Balogh. On k-coverage in a mostly sleeping sensor network. In *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 144–158, New York, NY, USA, 2004. ACM Press.
- [12] B. Liu and D. Towsley. A study of the coverage of large-scale sensor networks. In *Proceedings of the 1st IEEE Int. Conf. on Mobile Ad-hoc and Sensor Systems, MASS*, 2004.
- [13] M. Liu and C. Hsin. Network coverage using low duty-cycled sensors: Random and coordinated sleep algorithms, 2004.
- [14] F. G. Nakamura, F. P. Quintao, G. C. Menezes, and G. R. Mateus. An optimal node scheduling for flat wireless sensor networks. In *ICN 2005, LNCS 3420*, pages 4596–466, 2005.
- [15] G. Simon, M. Molnár, L. Gönczy, and B. Cousin. Robust k-coverage algorithms for sensor networks. *Special Issue of the IEEE Transactions on Instrumentation and Measurement*, August 2008.
- [16] G. Simon, P. Völgyesi, M. Maróti, and A. Lédeczi. Simulation-based optimization of communication protocols for large-scale wireless sensor networks. In *IEEE Aerospace Conference*, 2003.
- [17] G. Xing, X. Wang, Y. Zhang, C. Lu, R. Pless, and C. Gill. Integrated coverage and connectivity configuration for energy conservation in sensor networks. *ACM Trans. Sen. Netw.*, 1(1):36–72, 2005.