prof. dr hab. inż. Janusz Górski
mgr inż. Alan Turower
Department of Software Engineering
Faculty of Electronics, Telecommunications and Informatics
Gdańsk University of Technology
jango@eti.pg.gda.pl
alan.turower@eti.pg.gda.pl

Assessing the time effectiveness of trust management in fully synchronised wireless sensor networks

Abstract
The paper presents the results of the time effectiveness assessment of the distributed *WSN Cooperative Trust Management Method - WCT2M* in a fully synchronized Wireless Sensor Network (WSN). First we introduce some basic types of synchronization patterns in WSN based on the idea of *sleep scheduling*. Then we explain how WCT2M works in the network applying the fully synchronized sleep scheduling pattern. Such networks were subjected to the analyses with the help of a dedicated WCT2M simulator to investigate the time delays needed to identify and isolate the network nodes which depart from the assumed behavioural characteristics. The results of these simulations are presented to demonstrate the time effectiveness of WCT2M.

Keywords: sleep scheduling, trust management, simulations, time effectiveness

Introduction

Wireless sensor networks (WSN) increase their role in the application areas with high dependability expectations, including healthcare, transport, environment monitoring and others. Sensor nodes are distributed, often difficult to access, exposed to adverse environmental conditions, vulnerable to intentional attacks and subjected to severe limitations of their resources. This increases the possibility that some nodes can departure from the agreed policies that may adversely affect the mission fulfilled by the whole network.

To deal with this problem we proposed a distributed trust management method for WSN [1,2,3] which we call *WSN Cooperative Trust Management Method* - *WCT2M*. This method supports mutual assessment of the trustworthiness of the nodes and in consequence detection and isolation of the nodes that exhibit abnormal behaviours.

The objective of this paper is to examine the time effectiveness of WCT2M for a network applying a chosen synchronisation pattern. First we review the basic types of synchronisation in WSN and in particular present the idea of sleep scheduling. Next we describe how WCT2M can be applied for a network observing the fully synchronized sleep scheduling pattern. Then we present the results of experiments performed with the help of a simulator of WCT2M which we are presently developing in our laboratory.

1.1. Related works

Kumar et al. [4] presented a comprehensive survey on scheduling algorithms for *Time Division Multiple Access (TDMA)* protocol. They conclude that there is no a single protocol accepted as a standard. One of the reasons for this is that the *Media Access Control (MAC)* protocol choice is, in general, application dependent, which means that there is no one standard MAC for sensor networks. Another reason is the lack of standardization at lower layers (physical layer) and the (physical) sensor hardware.

Keshevarzian et al. [5] described main sleep scheduling patterns. Patterns differ in shift between wakeups. It allows minimizing the delays or power consumption in the network, depending on the network destination and type of traffic.

There were also some attempts to assess time effectiveness of trust management in WSN. In the experiments described by Zia [6] each WSN node transfers one packet every n seconds. Then the time needed to detect all distrusted nodes is used to assess effectiveness of the method.

Maroti et al. [7] examine the number of errors in a period of time and create histograms using real time of simulation.

Loscri et al. [8] present in their experiment results numbers of nodes or amounts of data (sent / received) in a period of time.

All this methods differs in the way a trust level is calculated and propagated among other nodes. Our approach allows for distributed and cooperative trust level calculation on all nodes in the network.

1.2. Sleep scheduling in WSN

Most of existing contention-based WSN MAC protocols reduce idle listening which is one of the most common sources of energy loss in WSN [9]. TDMA reservation based protocols establish fixed time periods for nodes to communicate to eliminate the channel contention and idle listening energy costs [10]. *Sensor MAC* (S-MAC) [11] and *Timeout MAC* (T-MAC) [12] are contention-based protocols focused on reducing idle radio listening by concentrating the network's data transmissions into a smaller active period and then transitioning to sleep for the remainder of the cycle. *An energy efficient and low latency MAC* DMAC [13] is an efficient data gathering protocol for sensor networks where the communication pattern is restricted to an unidirectional tree.

In the Zigbee stack, there is a fixed wakeup/sleep scheduling method: in each cycle, the nodes wake up twice, firstly to receive packets from their children and secondly, to transmit to their parents (children) in a Zigbee beacon-enabled tree network [14].

All these protocols allow creating *sleep scheduling* (*wakeup patterns*). This method provides for effective communication and for power usage minimization. Researchers in ad-hoc and sensor networks continue to search for new wakeup patterns to save power without suffering the large latency penalties associated with the wakeup process. Current methods can be divided into two main categories [5]:

- *Scheduled wakeups*: in this class, the nodes follow deterministic (or possibly random) wakeup patterns. Time synchronization among the nodes in the network is generally assumed. However, asynchronous wakeup mechanisms which do not require synchronization among the different nodes are also categorized in this class. Although asynchronous methods are simpler to implement, they are not as efficient as synchronous schemes, and in the worst case their guaranteed delay can be very long.
- *Wakeup on-demand* (*out-of-band wakeup*): It is assumed that a node can be signalled and awakened at any point of time and then a message is sent to the node. This is usually implemented by employing two wireless interfaces (radio receivers). The first radio is used for data communication and is triggered by the second ultra low-power (or possibly passive) radio which is used only for paging and signalling. Although these methods can be optimal in terms of both delay and energy, they are not yet practical. The cost issues, currently limited available hardware options which results in limited range and poor reliability, and stringent system requirements prohibit the widespread use and design of such wakeup techniques.

To use scheduled wakeups there is a time synchronization protocol needed, because time synchronization between all nodes is crucial. It is essential that all the nodes are able to wake up at the same time to be able to exchange information [15].

1.3. The case study

Let us consider a network with the distributed trust management model WCT2M [1,2,3] implemented. The network is given in Figure X.1, where BS is the *base station* (the *sink*) and all nodes are numbered. The *level* denotes the number of hops from a node to the base station. The maximal level is denoted N and in the considered network N = 2. The nodes with the level < N are *routers* – they forward messages from the sink to their child nodes and from their child nodes to the sink.

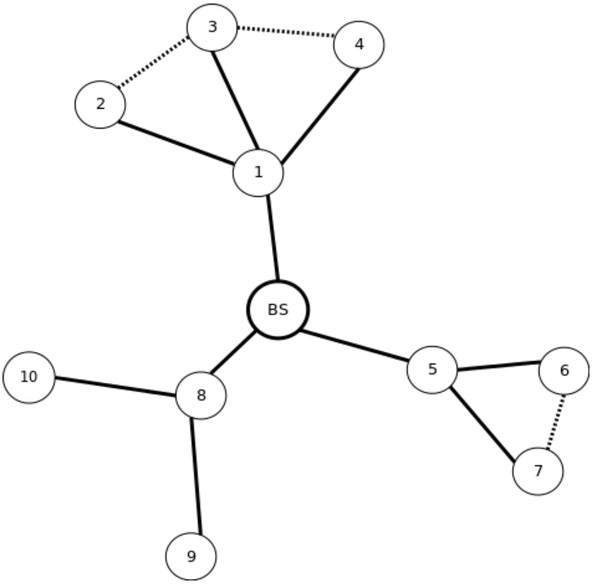

Fig. X.1. An example network with N=2

Source: own

We assume the network applies the *Fully Synchronized* sleep pattern [5] presented in Figure X.2.

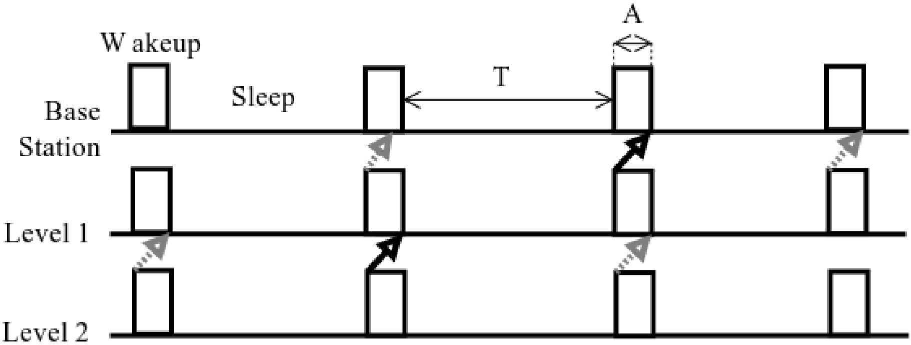

Fig. X.2. Fully Synchronized sleep pattern

Source: developed on the basis of [5]

The period of the sleep scheduling pattern is denoted T and the period of wakeup is denoted A. During its wakeup a node can send and receive multiple messages. Each message received in a given wakeup period is processed and is forwarded in the next wakeup [5]. During one wakeup period:

- A node can send a message to the base station using route paths (solid lines).

- The base station can send a broadcast message to all nodes using route paths.
- A node can send a broadcast message to its neighbours (solid lines and dotted lines).

For the considered network we distinguish a period called *turn*, denoted S in Figure X.3. It represents the time needed to send a message from a highest level node to the sink or to receive the sink's answer sent back to the node[1]. In the network following Fully Synchronized pattern, S = N (T + A). Every subsequent turn is shifted from the previous one by T + A. In the network of Figure X.1, S = 2 (T + A).

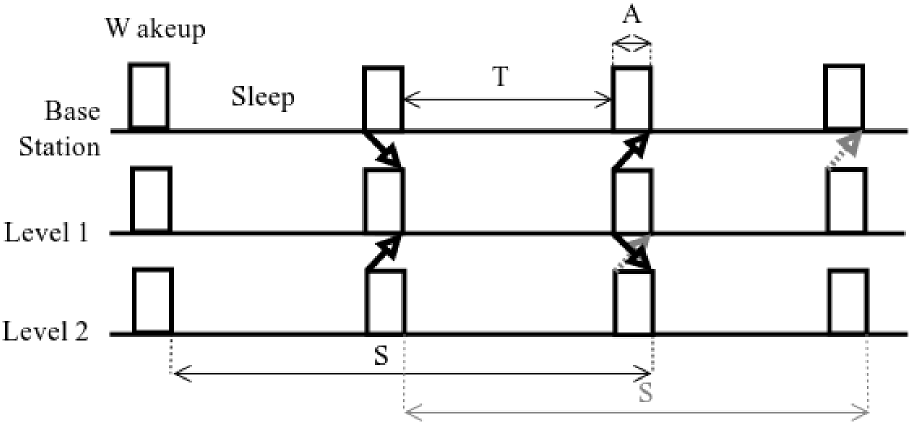

Fig. X.3. Fully Synchronized pattern

Source: own

Messages in the network can depart from their expected behaviours. For instance, a message can be spoiled during transmission (e.g. due to interferences), a malicious node can spoil a message intentionally and so on. In a network running WCT2M, each message is being evaluated by the trust mechanism [1] and the result of this assessment influence the trust attributed to the sending node. For instance, if a spoiled message is not detected by a router node and is forwarded to the next-hop node which detects the fault, the detecting node lowers its trust in the router node.

[1] Maximal time contains one T period of waiting for a wakeup period on the source node.

In the network running WCT2M, every node has the initial *trust level* in the eyes of its neighbours. This trust may change while the nodes communicate with each other. The trust level is represented in the scale <0;100> and its current value represents the node's reputation. Each node holds a *trust table* (a data structure where the trust levels of other nodes are represented). The trust tables can be exchanged (broadcasted) between the neighbouring nodes during each wakeup and the trust tables of each node are being updated accordingly. If the trust to a neighbouring node drops below the *cut off point*, the node stops to communicate with the neighbour. It eventually results that a node is *cut off* from the network, if any node on every route path from this node to the sink considers it as untrustworthy.

Simulation experiments

To learn more about the behaviour of the WCT2M trust management mechanism we have developed a dedicated simulator. The simulator takes the following input parameters which characterize the simulated network.

Table X.1. Input variables of the simulator

Input Variable	***Description***	***Scale***
p_n	Probability of sending a message by a node in a turn (*nodes activity*), also probability of sending the trust table to neighbours in a turn.	0-100%
p_b	Probability of sending a message by the sink in a turn (*sink activity*).	0-100%
p_s	Probability of spoiling a message by a malicious node.	0-100%
r	Probability of recognizing a spoiled message by a receiver node.	0-100%
e	Probability of spoiling a massage during transmission.	0-100%
t_T	Duration of the sleep scheduling pattern T.	[ms]
t_A	Duration of the wakeup period.	[ms]
t_S	Duration of the turn.	[s]
l_i	Initial trust level.	0-100
l_c	Cut-off point.	0-100

With respect to the network presented in Figure X.1 we conducted three experiments to learn about WCT2M time effectiveness in detecting and isolating faulty nodes, differently distributed in the network topology. The experiments are characterized in the table below.

Table X.2. Characteristic of the experiments

Experiment number	***Experiment characteristics***
Experiment I	the node labelled ‘1’ in Fig. X.1 is faulty (a router node)
Experiment II	the node labelled ‘2’ in Fig. X.1 is faulty (a leaf node)
Experiment III	the nodes labelled ‘1’ and ‘2’ in Fig. X.1 are faulty (a router node and a leaf node)

The simulation input parameters used during these experiments are presented in Table X.3.

Table X.3. Input parameters for the experiments

Input Variable	***Value***
p_n	50-100%
p_b	50-100%
p_s	70%
r	90%
e	2%
t_T	900ms
t_A	100ms
t_S	2s
l_i	50
l_c	10

We assume that in the considered network every node has the same probability p_n of sending a message in a given turn. We also assume that the input variables do not change while the network is operating. In real networks these parameter values vary due to the network destination, the need to save energy and technical specification of sensors. The chosen values were used in the health care case study [3].

1.4. Simulation results

The simulation results are presented in Figures X.4. – X.6. Each figure presents how the number of turns needed to detect and cut-off the faulty nodes depends on the *nodes activity* p_n and *sink activity* p_b.

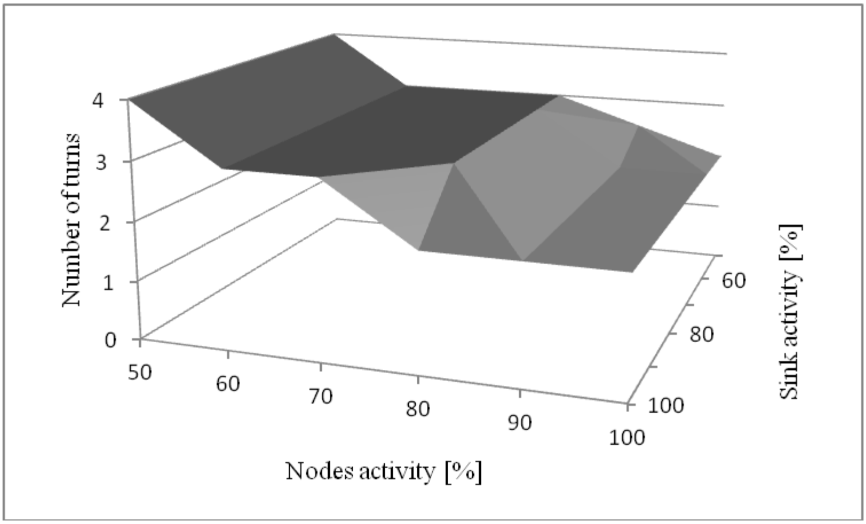

Fig. X.4. Experiment I results
Source: own

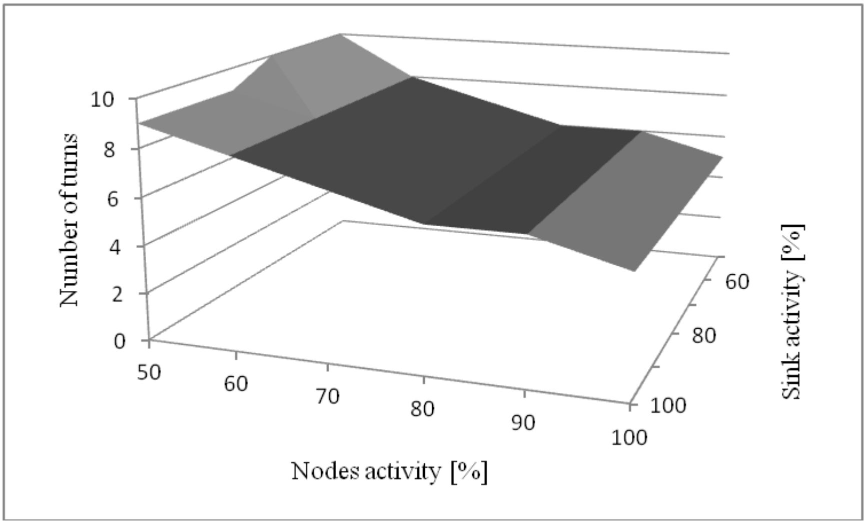

Fig. X.5. Experiment II results
Source: own

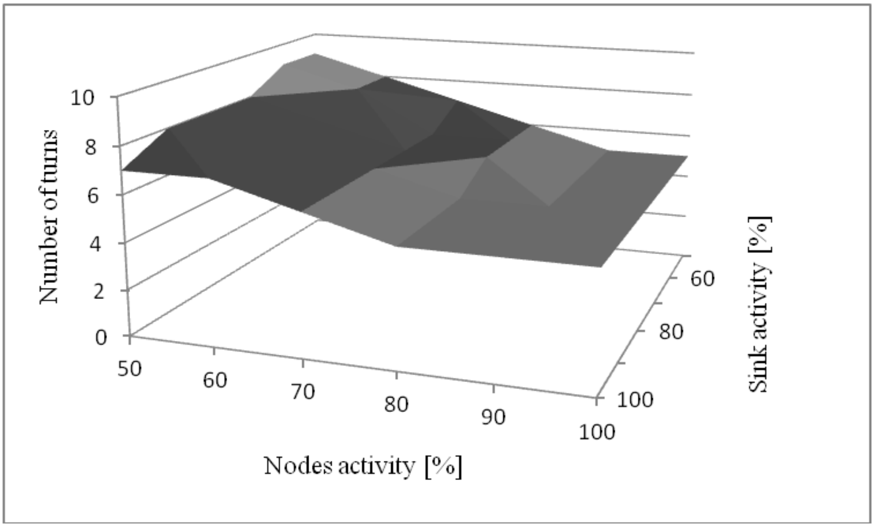

Fig. X.6. Experiment III results
Source: own

Conclusions

The experiments show that WCT2M mechanism can detect and cut off a faulty node in a relatively short time (3s in situation where the nodes are highly active and a router node fails). The detection time increases inversely to the nodes activity and for leaf nodes is longer than for router nodes.

The experiments also demonstrate that the time needed to detect and isolate a single faulty node depends mainly on the *nodes activity* and is less dependent on *sink activity*. Nevertheless the results demonstrate that sink activity increases its influence if we have multiple faulty nodes in the network.

In further research we plan to investigate networks of bigger size and with more complex patterns of faulty nodes distribution. In the scope of our research is also investigation of networks with mobile nodes.

Bibliography

1. Górski J., Turower A., Wardziński A.: *Distributed Trust Management Model for Wireless Sensor Networks*, Sixth International Conference on Dependability and Computer Systems DepCoS-RELCOMEX, 2011
2. Górski J., Turower A.: *Two-tier distributed trust management model for wireless sensor networks*, Forum Innowacji Młodych Badaczy, 2011
3. Górski J., Turower A.: *Trust management in WSN – case study evaluation*, ICT Young, 2012
4. Kumar S., Chauhan S.: *A Survey on Scheduling Algorithms for Wireless Sensor Networks*, International Journal of Computer Applications, No 5, 2011
5. Keshavarzian A., Lee H., Venkatraman L.: *Wakeup Scheduling in Wireless Sensor Networks*. Proceeding of the 7th ACM International Symposium on Mobile ad hoc networking and computing. 2006
6. Zia T. A.: *Reputation-based Trust Management in Wireless Sensor Networks*, International Conference on Multimedia and Ubiquitous Engineering, 2007
7. Maroti M. et al.: *The Flooding Time Synchronization Protocol*, proc. of 2nd ACM Conference on Embedded Networked Sensor Systems, 2004
8. Loscri V. et al.: *A Two-Levels Hierarchy for Low-Energy Adaptive Clustering Hierarchy (TL-LEACH)*, Vehicular Technology Conference, 2005
9. Brownfield M. et al.: *Wireless Sensor Network Energy-Adaptive MAC Protocol*, IEEE CCNC, 2006
10. Rajendran V., Obraczka K., Garcia-Luna-Aceves J.: *Energy-efficient MAC: energy-efficient collision-free medium access control for wireless sensor networks*, SENSYS, 2003
11. Wei Y., Heidemann J., Estrin D.: *An energy-efficient MAC protocol for wireless sensor networks*, INFOCOMM, 2002

12. van Dam T., Langendoan K.: *Energy-efficient MAC: An adaptive energy-efficient MAC protocol for wireless sensor networks*, ACM SENSYS, 2003
13. Lu G., Krishnamachari B., Raghavendra C.: *An Adaptive Energy-Efficient and Low-Latency MAC for Data Gathering in Wireless Sensor Networks*, IEEE Proceedings of the 18th International Parallel and Distributed Processing Symposium, 2004
14. Incel O., Ghosh A., Krishnamachari B.: *Scheduling Algorithms for Tree-Based Data Collection in Wireless Sensor Networks*, Theoretical Aspects of Distributed Computing in Sensor Networks, Springer, 2011
15. Sichitiu M., Veerarittiphan C.: *Simple, Accurate Time Synchronization for Wireless Sensor Networks*, IEEE Wireless Communications And Networking, Vol. 2, 2003