Implementation of an Adaptive Synchronizing Protocol for Energy Saving in Wireless Sensor Networks

Dragoş Mihai OFRIM*, Dragoş Ioan SĂCĂLEANU*,
Bogdan Alexandru OFRIM* and Rodica STOIAN*
*University „POLITEHNICA",
Faculty of Electronics, Telecommunications and Information Technology,
Bucharest, Romania
dragosofrim@yahoo.com, dragos_sacaleanu@yahoo.com, bofrim@yahoo.com, rodicastoian2004@yahoo.com

***Abstract* – In this paper we present an efficient protocol intended to save energy inside wireless sensor networks, thus increasing their lifetime. We offer a detailed description of the network model, topology, energy management and data synchronization to properly evaluate the performance of the proposed Adaptive Synchronizing Protocol. We simulated the proposed protocol and measured the network lifetime which proved to be significantly improved compared to the lifetime of the networks using a Non-Adaptive Synchronization model.**

***Index Terms* - wireless sensor network, energy saving, increase lifetime, adaptive synchronizing**

I. INTRODUCTION

Wireless sensor network is the universal name that defines a wide range of measuring devices. In essence, all the devices from this category have a local processor, a sensing device, local memory, are organized in an ad-hoc network and have wireless communication capabilities.

In designing a sensor network, the main challenge is to build systems that will run unsupervised for a long time [1]. This means, among others, that they need to have competitive hardware and software components, but most of all they need to have a well designed energy management, since that represents a limited resource. The present generation of sensor nodes uses battery as power source, which implies major constraints; newer generations use alternative energy sources like sunlight or vibrations, but they generate low currents, which builds the necessity of reducing the energy consumption [2].

The main causes of undesired energy loss in sensor networks are: transmission power, collisions, overhearing, idle listening, control packages, over processing of nodes. The solutions for eliminating or reducing this energy loss stay in: better routing protocol, network design and topology, nodes' synchronization and scheduling, having management and control capabilities implemented in sensor nodes [3], [16].

We have combined topology and routing improvements with management, synchronization and scheduling techniques in creating a sensor network model with an increased lifetime that provides the same performance and integrity of data.

II. RELATED WORK

Sensor networks are a continuous theme of intensive research, as better and better performances are searched. Most researches are made in the optimization of energy consumption. There is not a best solution for energy efficiency, as many solutions differ from a network type to another, from an application to another. A network can be periodic (nodes send data to the base station at a fixed interval) or event-driven (nodes send data to base station when an event occurs) [4], nodes can be location aware or not [5], [17], fixed or mobile, the communication can be unicast, multicast or broadcast etc [6]. Energy efficient algorithms have been developed in many integrating parts of a sensor network. Efficient routing protocols [7] like *LEACH* (Low Energy Adaptive Clustering Hierarchy) and *PEGASIS* (Power Efficient Gathering in Sensor Information System), which are developed in the network layer of the protocol stack, organize the nodes in clusters where the routing is made between individual nodes and their elected cluster head, which forwards the data to the base station. Regarding the MAC [8] (Medium Access Control), implemented in the data link layer, which handles the communication at one hop distance inside a network (scheduling of the sleep, receive and transmit states), several energy efficient algorithms have been successfully developed [9]: contention-based protocols like *LPL and Preamble Sampling*, in which nodes can start a transmission at any random moment and must contend for the channel, slotted protocols like *Sensor-MAC* in which the synchronization between nodes is establishes. Also, for networks with energy constrains where the location of the nodes must be established, localization algorithms, like *Ad-hoc positioning* by Niculescu and Nath, are implemented, as using GPS solutions in energy consuming.

III. KEY CONCEPTS AND TECHNIQUES FOR THE WIRELESS SENSOR NETWORK MODEL

In our development of the network model we have approached several concepts and techniques that are very important in designing an energy efficient sensor network.

A. Data centric and data aggregation

Since sensors are deployed in adjacent areas where they monitor common phenomena, it can be registered some redundancy in the data being communicated by the sensor nodes to the base station. A powerful solution to this problem is data aggregation, used for wireless routing in sensor networks. The essence of this concept is to combine the data sent by different sensor nodes on the same route, thus reducing the number of transmissions, eliminating redundancy and saving energy. This way, routing in sensor networks changes from being *address-centric* (developing the shortest routes between two addressable sensor nodes) to being *data-centric* [10] (developing routes from multiple sensor nodes to a base station that performs consolidation of redundant data).

To achieve data aggregation, the method of duplicate suppression could be used. If two sensor nodes generate the same data, the node that receives the incoming packets will send forward only one of them. In addition, nodes can perform additional aggregation functions like *max*, *min*, *avg* or any other functions with multiple input data.

Depending on the network topology, of the nodes' assignments and functions, the most commonly used data aggregation techniques are [14]:

A.1 In-Network Data Aggregation

The basic scheme of in-network data aggregation is shown in Figure 1. Each sensor device inside the region, after it detects the event, transmits its signal strength only to its neighbors. If the neighbor has a bigger signal strength, the sender remains silent and stops transmitting packets. Otherwise, it waits for packets from other sensors and, after receiving packets from all its neighbors, if the sender has the highest signal strength, it will then become the data aggregator and all other sensor devices stop detecting the event and helps only in routing the packet to the sink nodes.

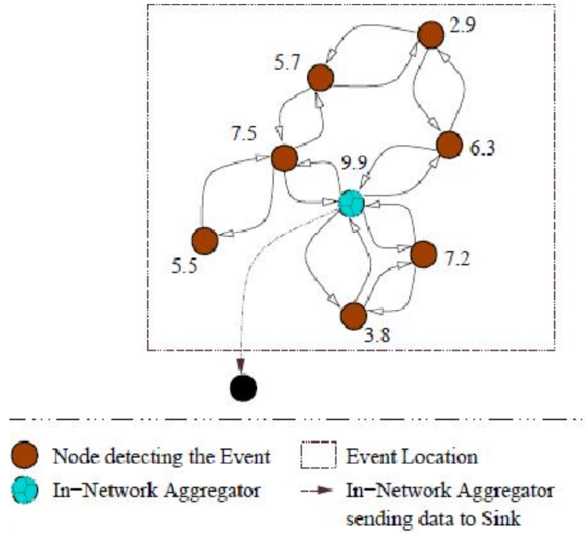

Figure 1. In-Network Data Aggregation

A.2 Grid-based Data Aggregation

Grid-based Data Aggregation technique is being used in mobile environments where the time duration of the events is very small. Such scenarios are being met in sensor network applications like military surveillance, weather forecasting, etc. In a classic topology, the sensor network environment is divided into a pre-defined set of grids or regions.

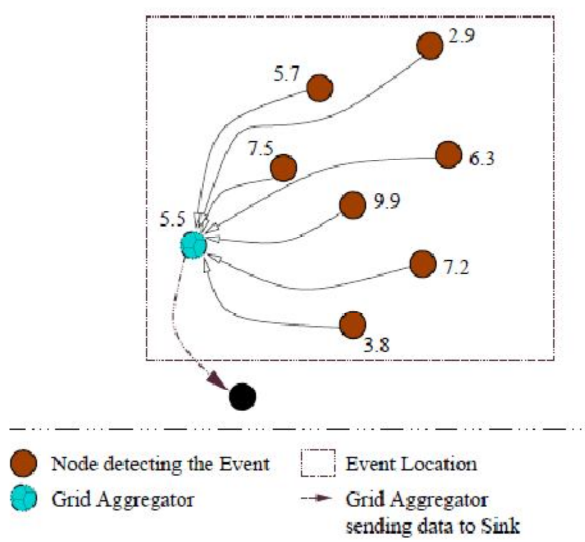

Figure 2. Grid-based Data Aggregation

In each region, the sensor nodes observe and report to the sink nodes events that occur inside the region. In addition, based on geographical position with respect to either the sink or the center of the grid, a sensor node is chosen as data aggregator. All other sensors inside the gird are informed about the location of the data aggregator and, during event detection, are sending the event information to it. After receiving all the data, the data aggregator sends only the critical information to the sink node. The typical Grid-based data aggregation scheme is drawn in Figure 2. .

This technique adapts well in case of dynamic changes in the network topology and event mobility. If the event is highly mobile in nature, it can be observed that many packets are exchanged between the sensors inside the region. Once the packets are received by the aggregator, only the most important information is sent to the sink nodes. Thus, by using Grid-based scheme, the traffic reduces and the critical information is securely transmitted to the end nodes interested in the data. Moreover, the throughput in such environments is also increased.

A.3 Hybrid Data Aggregation

In environments where events are highly localized, the In-Network data aggregation is preferred over the grid-based scheme. Unfortunately, in many of the sensor network applications, neither of these schemes could be used. In this case, the focus is on the performance provided by each of the schemes. Thus, considering the advantages and disadvantages associated with each of the In-Network and Grid-based scheme, a hybrid approach has been developed, which took the best characteristics of both the previous approaches. The basics are shown in Figure 3.

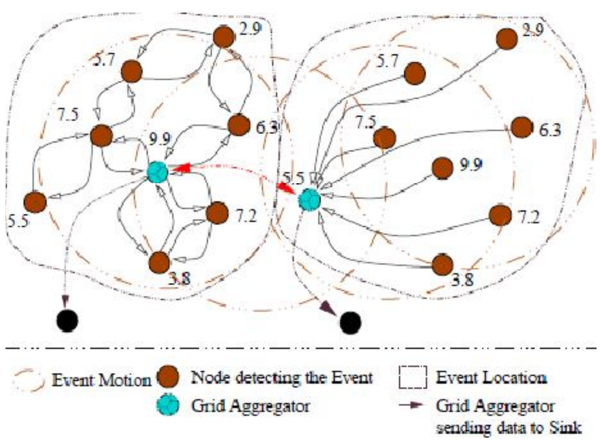

Figure 3. Hybrid Data Aggregation

As shown, at first every sensor is configured based on the In-Network scheme. When a sensor detects an event, its first action is to identify the sensor with the highest signal strength. In other words, the sensor which has the most critical and complete information about the event is identified, the same way as described in the in-network scheme. Moreover, a history is held by each node about past events and the corresponding signal strengths it detected. While event detecting, each node checks its table concerning the previous entry and tries to identify whether the event is highly mobile in nature or stationary. If it is determined that the event is localized, the in-network scheme is applied and an aggregator is chosen. On the other hand, if it occurs that there is a slow movement in the event, the sensor tries to send the information to the default aggregator

B. Mathematical Model for Energy Consumption

There are four main components of a sensor node: a radio transceiver, a microprocessor, a transducer and a battery. To model mathematically the energy consumption [11] of the first three components we need to define a set S of possible states $s_1,\ldots,s_k$ for each component. The total energy consumption of each component is calculated by summing the energy consumptions within each of the states $s_1,\ldots,s_k$ plus the energy needed to switch between states. We measure the energy consumption within a state s_j by a simple index n_j (which could represent the execution time or number of instructions), while the energy consumed to switch between states is calculated using a state transition matrix st, where st_{ij} represents the number of times the component switched from state s_i to s_j. We note as P_j the power needed in the state s_j and as E_{ij} the energy consumed for switching from state s_i to s_j. The total energy consumed by the component is:

$$E_{\text{consumed}} = \sum_{j=1}^{k}(n_j \times P_j) + \sum_{i,j=1,i\neq j}^{k}(st_{ij} \times E_{ij}) \quad (1)$$

For each component, the following states are defined:

Transceiver. There are four different states defined for the transceiver: *off, sleep, receiving* and *transmitting*. The transceiver consumes the most in the last two states, as the data is received or transmitted over the air. Thus, there is a need to reduce the time the radio communicates, meaning that the data packets have to be as short as possible and the radio in longer *sleep* or *off* states.

There are a variety of communication carrier frequencies that are used in the RF communication interfaces in wireless sensor networks. The most popular are: the 300–348MHz, 387–464MHz and 779–928MHz bands in the sub-1GHz spectrum and the 2.4–2.5GHz band. Considering the modulation techniques used for transmitting over the wireless environment, the most frequently used are: amplitude shift keying (ASK), binary frequency shift keying (2-FSK), binary phase shift keying (BPSK), frequency shift keying (FSK), Gaussian shaped frequency shift keying (GFSK), minimum shift keying (MSK), on-off keying (OOK), quadrature phase shift keying (QPSK) and orthogonal quadrature phase shift keying (O-QPSK).

For each of these carrier frequencies and modulation techniques, models may be defined for the properties of the transmitted signal over space. Examples of properties of interest include the *path loss*—the signal attenuation with distance—and the *bit error rate (BER)* for a given *signal to noise ratio (SNR)* or *signal to interference plus noise ratio (SINR)*. A simple model for the attenuation of signal strength, at distances d much larger than the carrier wavelength, defines the received signal power as proportional to

$$\frac{1}{d^2} \text{ (in free space)},$$
$$\frac{1}{d^4} \text{ (considering ground reflections)}.$$

To illustrate the theory above, the Friis free space model [15] defines the receive signal power, P_R for carrier wavelength λ, a receiver with receive antenna gain G_R, at distance d from a transmitter with transmit power P_T and transmit antenna gain G_T , as

$$P_R(d) = \frac{P_T G_T G_R \lambda^2}{(4\pi)^2 d^2}$$

Considering this model of energy consuming, the total amount of energy spent by the RF transceiver can be more accurately calculated, so better predictions of the lifetime of the sensor node can be made.

Microprocessor. There are four main states that can be identified for the microprocessor: *off, sleep, idle* and *active*. While in *sleep* mode, both the CPU and the most internal peripherals are off. The only routine available is the interrupt routine which sends the microprocessor into *idle* state. In the *idle* mode only some peripherals become active, such as the timer or the internal clock, while the CPU is still inactive. When switching to *active* state, the CPU and all peripherals become active. The energy consumption depends on the time the microprocessor has been in each state, the most energy consuming being the *active* state. Regarding the difference of energy consumption between radio and processor, both in *active* states, the radio consumes 100 to 1000 times more energy, no more data processing and algorithm implementation is preferred to long data packets.

Transducer. For a simple transducer, only two states are available: *on* and *off*. The transducer is active only a short period of time, until the ADC converter of the processor acquires the signal, so the sensor is not a big energy consumer.

Taking into consideration all these aspects, our network model reduces very much the amount of data transmitted over the air by increasing the local processing of data in every sensor node.

C. Time Division Multiple Access Scheduling Protocol

The MAC (Medium Access Control) protocols, implemented in the Data Link Layer, coordinate the sending-receiving actions inside a network, making the communication between several devices over a shared channel possible. These MAC protocols can be divided into: contention-based protocols (nodes can start a transmission at any random moment and must contend for the channel), slotted protocols (nodes have synchronized sleep periods, after which they wake-up and contend for the channel) and schedule-based protocols, in which the sleep periods and data transmission are scheduled. TDMA (Time Division Multiple Access) [12] is a schedule-based protocol and it is best fit for an energy efficient network.

The major advantages of this protocol are: it is collision free, needing no additional data transmission or acknowledge; idle listening is eliminated since nodes have a schedule when to expect incoming data. On the other hand, it needs increased processing capacity, which is less energy consuming than the additional radio transmission.

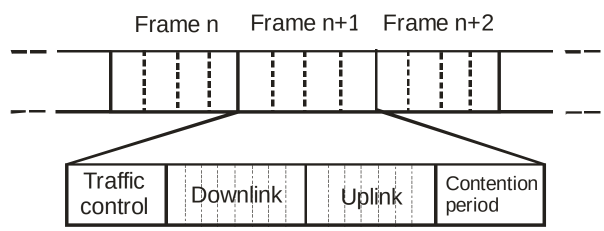

Figure 4. TDMA Protocol Frame Structure

TDMA protocol divides channels into slots, which are grouped in frames (Figure 4). The network coordinator schedules the nodes to use a certain slot. This can be done per each frame, or it can be repeated over several frames. In the first time slot of the frame, the traffic control information is broadcasted by the coordinator to all nodes in its cell, which informs each node when to be ready to receive a packet (down-link slot) and when it is allowed to send a packet (up-link slot). The last time slot of the frame is a contention period in which the registration of new sensor nodes is done by the network coordinator, which also schedules them for future communication.

There are several [11] ways to implement the TDMA protocol:

C1. Sink-based scheduling.

The main idea of this approach is to divide the network into large clusters, allowing multi-hop routing. Inside each cluster, the traffic is scheduled by a sink node connected to the wired backbone network. This architecture optimizes the network lifetime, as the sink takes into consideration the energy levels of the nodes and schedules which nodes will sense, route or sleep. The TDMA schedule is periodically adapted to changes. A prime request is that all nodes can communicate directly with the sink node, thus limiting the scalability.

C2. Static scheduling

The Self-Stabilizing protocol implies a fixed schedule throughout the lifetime of the whole wireless sensor network, removing the need of a distributed scheduler. This protocol is mainly used in basic topologies like square or hexagonal grids. The traffic is synchronized in rounds – in the even rounds the north-bound messages are transmitted, while in the odd rounds the south-bound messages are routed. The performance results of such a protocol are acceptable for typical communication patterns (broadcast, convergecast and local gossip), but its withdrawn refers to the constraints of the nodes' location.

C3. Rotating duties.

When the density of nodes is increased, the energy consumed by the access point can be amortized over multiple nodes by rotating duties principle. The network is organized into clusters connected by gateway nodes. At the start of each TDMA frame, nodes send a bit of information to their cluster head stating if they have data to send. Having this information, the cluster head calculates the data slots needed, assigns the slots and broadcasts this information to all the nodes under its control. After each TDMA window a new cluster head is elected based on the nodes' residual energy.

C4. Partitioned scheduling.

In this protocol, the scheduling duties are partitioned according to slot number. Each slot is considered as a small TDMA frame, formed by a contention phase, a traffic control section and a data transmission session. An active node that owns a slot transmits only in its own slot. Therefore, o node must listen to the traffic control sections of all its neighbors, since it may be the desired receiver of any of them. The contention phase is included for the use of the passive nodes that are not assigned a slot; to form a backbone network only some nodes need to be active. The passive nodes will use it when they detect an event. In many cases, events occur rarely, so listening for inexistent requests will lead to an undesired energy loss.

C5. Replicated scheduling.

The main idea of this protocol is to replicate the scheduling process over all nodes from a network. Nodes always broadcast information about the traffic routed by them and the identities of their one-hop neighbors. Thus, each node is informed about the demands of its one-hop neighbors and the identity of its two-hop neighbors. With this knowledge it is easy to determine a collision-free slot assignment by means of a function that computes the winner of each slot based on the node identities and slot number. As the execution evolves, the schedule is updated to match the current traffic conditions, nodes with little traffic may release their slot for the remainder of the frame for use by other nodes.

D. Code Division Multiple Access

CDMA is a spread spectrum multiple access protocol. A spread spectrum technique spreads the bandwidth of the data uniformly for the same transmitted power. In CDMA, a locally generated code by the transmitting entity runs at a much higher rate than the data to be transmitted. The data that needs to be transmitted is simply XOR added with the faster code. In the Figure 4 it is shown how spread spectrum signal is generated.

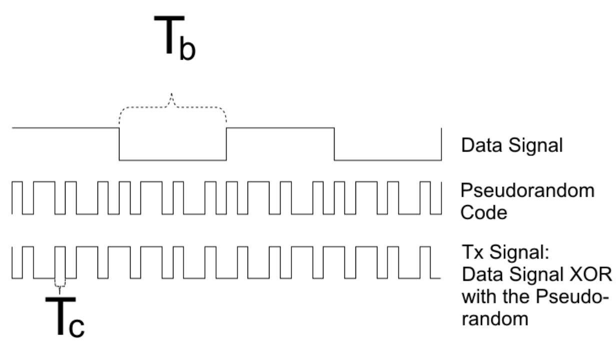

Figure 5. Generation of CDMA

The data signal to be sent has the pulse duration of T_b.. It is XOR added with the code signal with pulse duration of T_c. The bandwidth of the data signal is 1 / T_b, while the bandwidth of the spread spectrum signal is 1 / T_c. Since T_c is considerably smaller than T_b, the bandwidth of the spread spectrum signal will be much larger than the bandwidth of the original signal. The ratio T_b / T_c determines the spreading factor or processing gain and defines to a certain extent the upper limit of the total number of users supported simultaneously by a base station [18]. To use a CDMA system, each user has a different code for signal modulation. Very important in the performance of a CDMA system is the choosing of the codes to modulate the signal. The best performance will be obtained by a good separation between the signal of a desired user and the signals of other users. At reception, the separation of the signals is made by correlating the received signal with the locally generated code of the desired user. If the signal matches the desired user's code then the correlation function will be high and the system will be able to extract that signal. If the desired user's code has nothing in common with the signal, the correlation should be as close to zero as possible (thus eliminating the signal) and it will be interpreted as a random noise; this is referred to as cross correlation.

IV. DESCRIPTION OF WIRELESS SENSOR NETWORK MODEL

Implementing the techniques mentioned above will consistently reduce the energy consumption of a sensor network. To reduce even more the energy consumption without affecting the performance and integrity of data, our solution is to use the adaptive scheduling algorithm. The main idea is that when several sensor nodes from the same area provide redundant data, the one whose battery has the most energy left should keep its schedule, while the other sensor nodes receive bigger sleep periods. After all nodes become active again, a new evaluation is being made. This comes in addition to data aggregation technique, which is used when all nodes are active. We provide now a detailed description of the network and algorithm.

A. Network topology and routing

The network is organized in clusters [19], each with a self elected cluster head (Figure 6). Each sensor node transmits the data to its cluster head (CH) which forwards it to the base station. The CH implements both the adaptive scheduling and the data aggregation within its cluster. Using clusters, lots of energy is saved, as the sensor nodes only send data to their CH and not far away to the base station. In this situation, the CH consumes the biggest amount of energy. To prevent the loss of the CH, it periodically changes, new CHs being elected based on energy consideration. This way, the energy consumption is homogenized inside the cluster and the lifetime of the network is increased.

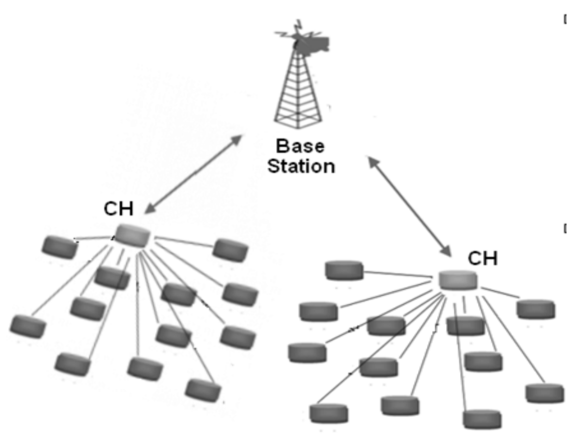

Figure 6. Cluster based topology

B. Network communication

Inside the clusters, the communication between the sensor nodes and CH is implemented using TDMA protocol. For the clusters not to interfere which each other, CDMA (Code Division Multiple Access) [13] is used. Before transmission, the data packets from the sender are XOR-ed with a specific code. At reception, the receiver XOR-es the data packets with the same code, to retrieve the original data. Receivers using another code to decode the message perceive the transmission as a random noise. Each cluster has a different code, thus preventing interference of data transmission.

C. Network devices

The network is formed by the following types of devices, each with a well defined role:

Sensor nodes. The sensor nodes deployed may be hundreds or even thousands. Their role is to sense the environment, process the data and forward it to the cluster head. Each sensor node has the processing capability to be a cluster head.

Cluster heads. The cluster heads are coordinators – they organize the communication inside each cluster, the data transmission, the sleep periods, the data aggregation and they forward the processed data to the base station. When they are low on energy, another cluster head is elected from the sensor nodes.

Base station. The base station is the destination of the measurements, usually a PC with storage and visual display capabilities, accessed by the human operator.

V. ADAPTIVE SYNCHRONIZING PROTOCOL

In the first step, the clusters are being created. There is a certain number of cluster heads which are elected from the nodes with the biggest amount of energy available. After that, the remaining nodes elect the cluster to be a part of depending on the SNR (Signal to Noise Ratio) measured during the communication with every CH. The clusters being formed, each CH implements the CDMA and TDMA protocol, scheduling the slots in which the sensor nodes send and receive data.

As mentioned before, the sensor nodes with redundant data are put by the CH in bigger sleep periods. It is vital to define all the time frames used in this matter. An active sensor node has three important time intervals:

- Measuring time – *Tm* – the interval in which it senses the environment
- Processing time – *Tp* – the interval in which it processes the measured data
- Transmitting time – *Tx* – the time required for transmitting data to the CH

- Receiving time – *Rx* – the time required for receiving data from the CH

Additional to this active period (*Ta*) is the sleep interval, in which the sensor node does nothing (*Ts*). To reduce even more the energy consumption at the node level, the functioning sequences of the nodes are delayed (Figure 3), as they are scheduled individually for data transmission. The active period combined with the sleep period for all the nodes in the cluster determines the *Pa* period of the cluster:

$$Ta = Tm + Tp + Tx \qquad (2)$$
$$Pa = Tm + Ts \qquad (3)$$

The sensor nodes that are assigned longer sleep periods due to their redundant data are referred to as functioning at a *Pb* interval. For them to synchronize with the nodes functioning at a *Pa* interval, *Pb* must be:

$$Pb = Ta + Tsb \qquad (4)$$
$$Tsb = b * Ts + (b-1) * Ta \qquad (5)$$

Tsb is the increased sleeping period of the nodes functioning at a *Pb* interval, while *b* is a constant which determines the length of the sleeping period. Depending on the variation factor of the measured physical parameter, *b* can be higher (if the parameter changes slowly) or smaller (if the parameter changes fast). It can be observed that *Pa* is a particular case of *Pb* – when *b* equals 1, *Pa* equals *Pb*

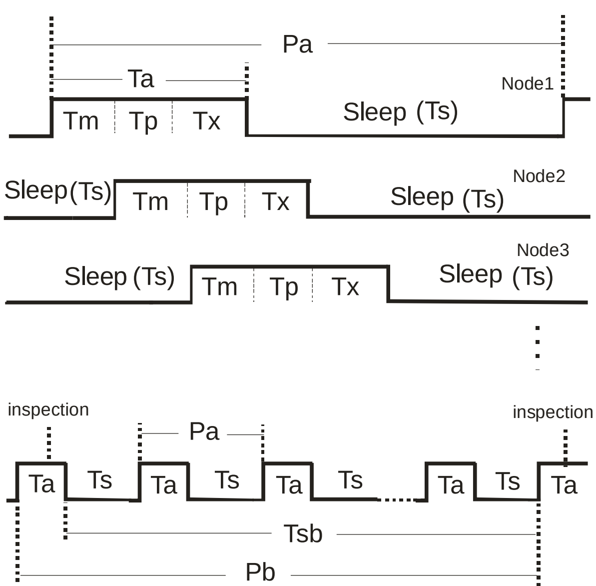

Figure 7. Time frames of the sensor nodes

During *Tsb*, the nodes functioning at a *Pa* interval send data to the CH, while the CH only aggregates the data and forwards it to the base station. During each *Ta* of *Pb*, all nodes are active and send measured data to their CH.

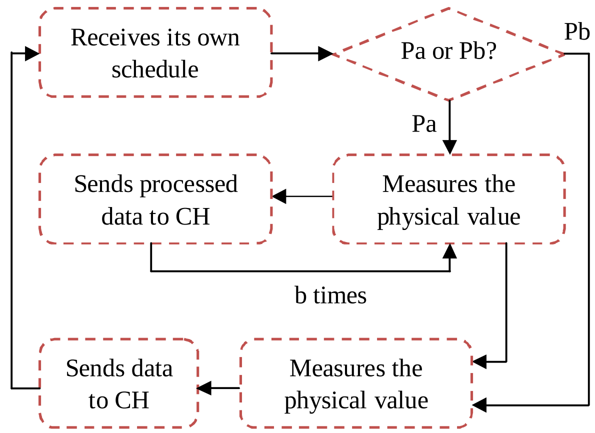

Figure 8. Sensor node's action sequence

The action sequence of the sensor nodes is presented in Figure 8.

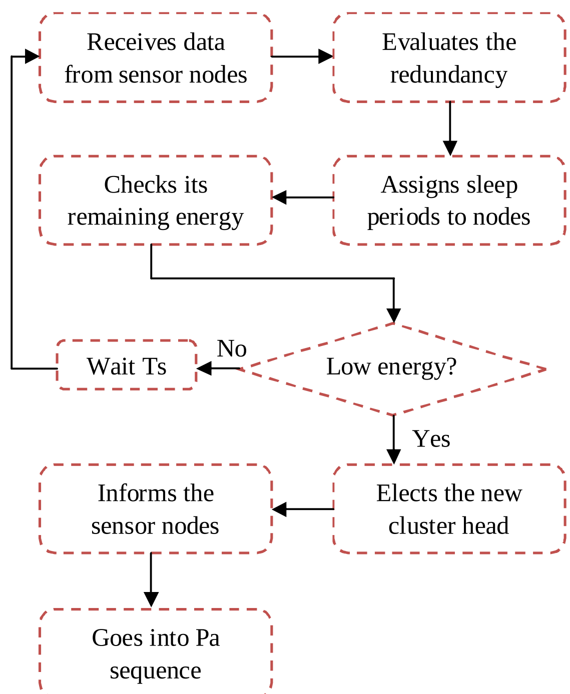

Figure 9. CH's action sequence

At the end of the *Pb* period, when all the data are gathered, the CH performs two tasks:

- evaluates the redundancy of the received data and establishes which sensor nodes will have a *Pa* sequence and which a *Pb* sequence. When two nodes measure the same data, within a pre-established soft threshold *St*, the node with the biggest amount of energy left is assigned to function at a *Pa* interval, while the other will function at a *Pb* interval.

- checks its remaining energy; if the energy is too low, the sensor node from the cluster with the biggest amount of energy left is elected as the new CH. After this, the functioning sequence starts over with the new CH being responsible for the cluster management.

The action sequence of the CH is presented in Figure 9.

VI. SIMULATION RESULTS

To prove the efficiency of the adaptive synchronizing protocol, we have built a simulation program using OMNet++ platform. We have tested two network configurations, both having cluster topology, data aggregation technique and TDMA scheduling, but only one implementing the adaptive synchronizing protocol (ASP). The non-adaptive scheduling model (NASP) (when all nodes function at a *Pa* interval, *b* equals 1) is similar to the model used in defining other energy efficient algorithms like LEACH or PEGASIS. The measured network parameter was the network lifetime, depending on the number of nodes in the network and the length of the *Tsb* period.

$$T = f(n,b)$$

T – lifetime of the network (h)
n – number of sensor nodes in the network
b – constant in (5)

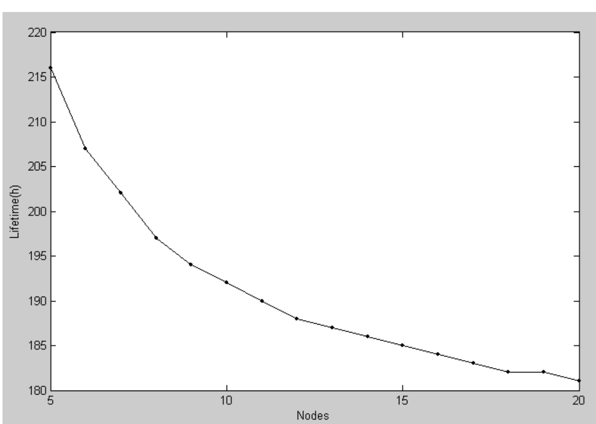

Figure 10. Lifetime of a NASP network

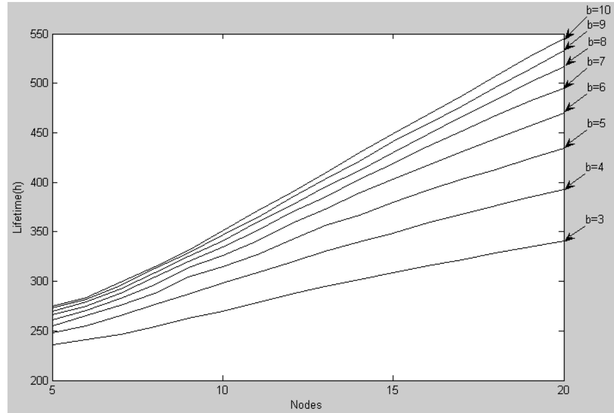

Figure 11. Lifetime of an ASP network

The simulations were made at a cluster level. The network lifetime is measured in hours and represents the moment when 30% of the nodes in a cluster run out of energy, affecting the network's correct functioning. The sleep interval *Ts* was chosen to be of 10s.

In case of NASP, the experimental results are shown in Figure 10. It can be observed that, as the number of nodes in a cluster increases, the lifetime of the network decreases, as the number of data packets to route increases.

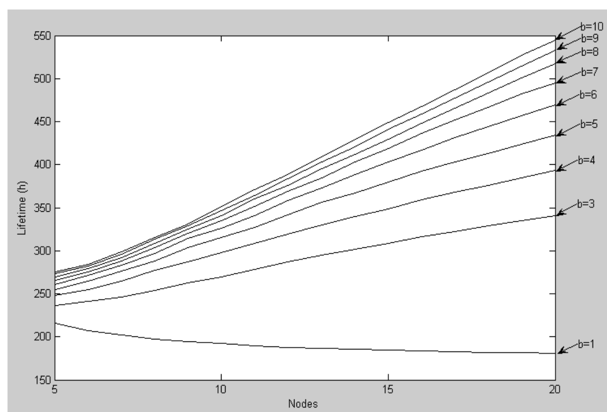

Figure 12. Lifetime of network comparison between ASP network and NASP network

In case of an ASP network, the experimental results are shown in Figure 11. The simulations were made for the same series of number of nodes as in the NASP case, but for different values of *b* (from 3 to 10). Different from NASP network, when applying the Adaptive Synchronizing Protocol, as the number of nodes increases the network lifetime increases. This is a fortunate consequence of applying ASP that stays in the elimination of the redundancy of data packets coming from the field nodes while more and more nodes with energy conserved are available for becoming Cluster Heads (which is the most energy consuming)

Unfortunately, the number of nodes in a cluster cannot be too high, as the nodes far away from the CH will consume more energy for the wireless communication. In this case, the solution is to form another cluster. Moreover, as an advantage, increasing *b*, which means increasing the sleep period of the nodes transmitting redundant data, the network lifetime also increases.

The difference in performance between ASP and NASP is illustrated in Figure 12. At any number of nodes, ASP is more competitive than NASP. The curve for *b* equal 1 represents the NASP network, while the curves for b going from 3 to 10 represent the ASP network.

VII. CONCLUSIONS

Combined with energy efficient schemes implemented in topology, data routing, scheduling and synchronization protocols, the adaptive synchronizing protocol offers an increased lifetime, reducing the energy consumption, a vital parameter in the functioning of sensor networks. For the same wireless sensor network model, the simulation results confirm an important increase in network lifetime in the case of applying the Adaptive Synchronizing Protocol compared to the network lifetime in the case of a Non-Adaptive Synchronizing model.

REFERENCES

[1] Estrin, D., Govindan, R., Heidemann, J. and Kumar, S. Next century challenges: scalable coordination in sensor networks. *In Proceedings of the International Conference on Mobile Computing and Networking*. ACM/IEEE Seattle, Washington, USA, 1999

[2] Akyildiz, I., Su, W., Sankarasubramaniam, Y., and Cayirci, E. A survey on sensor networks, *IEEE Communication Magazine*, 2002

[3] Slijepcevic Sasa, and Potkonjac, Miodrag. Power efficient organization of wireless sensor networks. In *Proceedings of the IEEE International Conference on Communications*, June 2001.

[4] Min, R., Bhardwaj, M., Seong-Hwan Cho, Shih, E., Sinha, A., Wang, A., and Chandrakasan, A. Low-power wireless sensor networks. In *Proceedings of the 14th International Conference on VLSI*

[5] Amitangshu Pal. Localization Algorithms in Wireless Sensor Networks: Current Approaches and Future Challenges. In *Network Protocols and Algorithms*, 2010, Vol. 2, No. 1

[6] Dragos Niculescu. Communication Paradigms for Sensor Networks. In *NEC Laboratories America, Princeton, NJ*, 2007

[7] Laiali Almazaydeh, Eman Abdelfattah, Manal Al-Bzoor, and Amer Al- Rahayfeh. Performance Evaluation of Routing Protocols in Wireless Sensor Networks. In *International Journal of Computer Science and Information Technology*, Volume 2, Number 2, April 2010

[8] Somnath Ghosh, Prakash Veeraraghavan, Samar Singh, and Lei Zhang. Performance of a Wireless Sensor Network MAC Protocol with a Global Sleep Schedule. In *International Journal of Multimedia and Ubiquitous Engineering* Vol. 4, No. 2, April, 2009

[9] http://www.isi.edu/ilense/macss/

[10] Bhaskar Krishnamachari, Deborah Estrin, and Stephen Wicker. Modelling Data-Centric Routing in Wireless Sensor Networks. In *Computer Engineering Technical Report CENG,* 2002

[11] Richard Zurawski. Embedded Systems Handbook. *CRC Press* 2006

[12] Rozeha A. Rashid, Wan Mohd Ariff Ehsan W. Embong, Azami Zaharim, Norsheila Fisal. Development of Energy Aware TDMA-Based MAC Protocol for Wireless Sensor Network System. In *European Journal of Scientific Research ISSN 1450-216X* Vol.30 No.4 (2009), pp.571-578

[13] Piergiuseppe di Marco. Protocol Design and Implementation for Wireless Sensor Networks. In *Masters' Degree Project Stockholm*, Sweden April 2008

[14] Karthikeyan Vaidyanathan, Sayantan Sur, Sundeep Narravula, Prasun Sinha, Data Aggregation Techinques in Sensor Networks, 2004

[15] Sathaporn Promwong, Pichaya Supanakoon, Near-field Characterization of a Free Space Channel Model for Ultra Wideband Radi, *In Proceedings of International Conference on Electromagnetic Compatibility*, Phuket, Thailand, 2005

[16] Georgios B. Giannakis, Distributed Estimation Using Wireless Sensor Networks, *In Proceedings of the 12th WSEASInternational Conference on Communications,* Heraklion, Crete, Greece, 2008

[17] Titsa Papantoni-Kazakos, Traffic Monitoring and Rate Allocation in Sensor Networks, *2nd WSEAS International Conference on SENSORS and SIGNALS*, Baltimore, USA, 2009

[18] H. Dehghan, Z. Zeinalkhani, S. M. Safavi, H. R. Amindavar and A. R. Moghaddamjoo Multiple target tracking in wireless networks based on time-frequency distributions, *In Proceedings of the 1st WSEAS International Conference on SENSORS and SIGNALS*, Bucharest, Romania, 2008

[19] Gianluigi Folino, Agostino Forestiero, Giandomenico Spezzano A P2P-based Flocking Algorithm for Distributed Clustering using Small World Structure *In Proceedings of the 7th WSEAS International Conference on APPLIED INFORMATICS and COMMUNICATIONS*, Athens, Greece 2007