An Integrated Model of Clustering and Routing in Wireless Sensor Networks

Arun Thangavel and Yuvarani B

Abstract: Nowadays, the applications of wireless sensor networks for industrial purpose have been rapidly increased. However, the main limitation is power consumption. As communication typically accounts for the major power consumption, the activity of the transceiver should be minimized, in order to prolong the network lifetime. To this end, this paper proposes an integration technique with sleep protocol for efficient power management along with an load balancing concept. This provides sleep schedules of nodes to match the network demands, even in time-varying operating conditions. In addition, it does not require any a-priority knowledge of the network topology or traffic pattern. Also the load balancing reduces the prolonged usage of a particular node. Additionally, the paper includes the efforts carried out on developing optimization techniques in the area of routing protocols for wireless sensor networks. Another approach to extend Wireless Sensors Networks (WSN) lifetime is to use mobile sinks to increase message delivery latency.

Keywords: **Data Acquisition, Energy, Load Balancing, Lifetime, Mobile Sinks, Message Delivery Latency, Sensor Nodes, Wireless Sensor Networks.**

I. INTRODUCTION

A wireless ad-hoc network is a decentralized type of network. The network is ad hoc because it is an backbone less network, such as routers in wired networks or access points in managed (infrastructure) wireless networks. Instead, each node participates in routing by forwarding data for other nodes, and so the determination of which nodes forward data is made dynamically based on the network connectivity. In addition to the classic routing, ad hoc networks can use flooding for forwarding the data. An ad hoc network typically refers to any set of networks where all devices have equal status on a network and are free to associate with any other ad hoc network devices in link range. Very often, ad hoc network refers to a mode of operation of IEEE 802.11 wireless networks. It also refers to a network device's ability to maintain link status information for any number of devices in a 1 link (aka "hop") range, and thus this is most often a Layer 2 activity.

This is only a Layer 2 activity; ad hoc networks alone may not support a routable IP network environment without additional Layer 2 or Layer 3 capabilities. The earliest wireless ad-hoc networks were the "packet radio" networks (PRNETs) from the 1970s, sponsored by DARPA after the ALOHA net project.

Arun Thangavel is with M.E- Applied Electronics, Department of Electrical Engineering, K. S. R. College of Engineering, Tiruchengode and Yuvarani B is working as Assistant Professor, Department of Electrical Engineering, K. S. R. College of Engineering, Tiruchengode. Emails: ecearun.india@gmail.com , yuvasasib@gmail.com

The decentralized nature of wireless ad-hoc networks makes them suitable for a variety of applications where central nodes can't be relied on, and may improve the scalability of wireless ad-hoc networks compared to wireless managed networks, though theoretical and practical limits to the overall capacity of such networks have been identified. The presence of dynamic and adaptive routing protocols enables ad-hoc networks to be formed quickly. Wireless ad hoc networks can be further classified by their application:

- Mobile ad-hoc networks (MANET)
- Wireless mesh networks (WMN)
- Wireless Sensor networks (WSN)

Wireless Sensor Network

A wireless sensor network (WSN) consists of spatially distributed autonomous sensors to monitor physical or environmental conditions, such as temperature, sound, pressure, etc. and to cooperatively pass their data through the network to a main location. The more modern networks are bi-directional, also enabling control of sensor activity. The development of wireless sensor networks was motivated by military applications such as battlefield surveillance; today such networks are used in many industrial and consumer applications, such as industrial process monitoring and control, machine health monitoring, and so on.

The WSN is built of "nodes"– from a few to several hundreds or even thousands, where each node is connected to one (or sometimes several) sensors. Each such sensor network node has typically several parts: a radio transceiver with an internal antenna or connection to an external antenna, a microcontroller, an electronic circuit for interfacing with the sensors and an energy source, usually a battery. A sensor node might vary in size from that of a shoebox down to the size of a grain of dust.

The cost of sensor nodes is similarly variable, ranging from a few to hundreds of dollars, depending on the complexity of the individual sensor nodes. Size and cost constraints on sensor nodes result in corresponding constraints on resources such as energy, memory, computational speed and communications bandwidth. The topology of the WSNs can vary from a simple star network to an advanced multi-hop wireless mesh network.

II.PROTOCOL DESCRIPTION

This section presents the description of ASLEEP protocol. After a general overview, we will describe the core components of the protocol, i.e., the sleep prediction algorithm (used by each node to dynamically estimate the length of its expected active period), and the sleep coordination algorithm (used to enforce the new sleep schedule throughout the network).Finally, we will introduce two mechanisms to improve the robustness of the protocol.

A.Protocol Overview

In the following we will refer to a data collection scenario where data typically flow from source nodes to the sink, while data from the sink to the sources are much less frequent. We will assume that nodes are organized to form a logical routing tree (or data gathering tree) rooted at the sink, and use an underlying CSMA (Carrier Sense Multiple Access) MAC protocol for communication. These assumptions are quite realistic, as most MAC protocols commonly used in WSNs are CSMA-based , and many popular routing protocols for WSNs rely on a routing tree. In a real deployment the routing tree is re-computed periodically to cope with possible topology changes and better share the energy consumption among nodes. However, as nodes are supposed to be static, we can assume that the routing tree remains stable for a reasonable amount of time. The communication between a parent and its children occurs in communication periods which repeat periodically. Each communication period includes an active interval during which nodes communicate by using the underlying MAC protocol, and a silence interval during which nodes turn their radio off to save energy. As shown in Figure 1, active intervals are staggered so that nodes at the lower levels in the routing tree wake up earlier than their ancestors. The active interval of each (intermediate) sensor node consists of two adjacent talk intervals (TI), the first one with its children and the other one with its parent2. Throughout, we will refer to the talk interval shared by a generic node j and all its children, during the m-th communication period.

Figure 1. Parameters of the sleep scheduling protocol.

During each communication period every parent node estimates the duration of the talk interval to share with its children in the next communication period by means of the algorithm B. Although parent nodes can independently set their talk interval, a collective effort is needed for the schedule of the whole network to remain consistent and energy efficient. Hence, as a result of a change in the talk interval of a single parent node, the network-wide schedule must be rearranged. This is accomplished by appropriately shifting active intervals of a number of nodes, so as to ensure that (i) the active intervals of all nodes are properly staggered, and (ii) the two talk intervals of each node are contiguous.

Two special messages, direct beacons and reverse beacons, are used for propagating schedule parameters to downstream and upstream nodes, respectively. Direct beacons are broadcast by every parent node to all its children during each communication period. Instead, reverse beacons are sent in the opposite direction, i.e., from a child to its parent. As it will be shown below, direct beacons are critical for the correctness of the protocol. Hence, ASLEEP also includes mechanisms for (i) increasing the probability of successful delivery of direct beacons, and (ii) enforcing a correct (even if non-optimal) behaviour of nodes in case they miss a direct beacon. In particular, to increase the probability of successful delivery, direct beacons are transmitted at the end of each talk interval, in a reserved time period (Beacon Period).

B.Talk Interval Prediction

In the ASLEEP protocol a sleep schedule is basically defined by the communication period and the talk interval of each individual parent node. The length of the communication period is closely related to the specific application and, thus, it is a global parameter specified by the sink when distributing the query. A variation in the communication period corresponds to a modification of the query, i.e. the new interval for the periodic data acquisition.

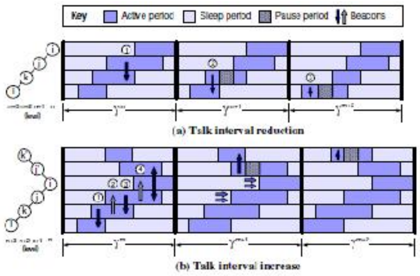

Figure 2.Examples of talk interval adaptation.

Choosing an appropriate talk interval is somewhat more involved. Ideally, each parent node should set the talk interval with its children to the minimum time needed to successfully receive all messages from all children. However, this time depends on a number of factors such as the underlying MAC protocol, channel conditions, degree of contention, and number of messages to be received, and so on. In its turn, the number of messages to be received by a sensor node depends on the number of its children, the message generation rate at source nodes, and the network topology. Therefore, computing the ideal talk interval would require the global knowledge of the network. Moreover this value should be continuously updated as the operating conditions change over time. Since such an approach is not practical, we propose here an adaptive technique that approximates this ideal scheme by letting every parent node to choose autonomously its own talk interval with its children. The decision involves only local information and, thus, it does not require knowing the network topology.

In principle, any algorithm can be used to estimate the expected talk interval in the next communication period. We used the simple algorithm discussed below. Each parent node measures and stores the following quantities.

- Message inter-reception time (Δ). This is the difference between the time instants at which two consecutive messages are correctly received.

- Number of received messages (npkt). The total number of messages correctly received in a single communication period.

A similar approach is employed also when a node increases the talk interval with its children. In this case, the node has to force its ancestors to defer their talk intervals, in order to accommodate the additional time required for communications. To this end, the node makes use of a reverse beacon, which is sent to its parent and forwarded up to the tree until the sink node is reached. Note that this step is required to ensure the correctness of the protocol, i.e. that t he talk intervals of intermediate nodes do not overlap. As above, the example depicted in Figure 2-b will help understanding. Suppose that node k at the (n+2)-th level of the tree decides to increase the talk interval with its children, i.e., nodes at the (n+3)-th level (including node l). First, node k advertises the new talk interval to its children through the direct beacon (step 1). Second, in the same communication period, the node sends the reverse beacon (step 2) to its parent j at the (n+1)-th level, to force a talk interval shift ahead in time. Node j receives the reverse beacon, adjusts the parameters for the next communication period and advertises them, via the direct beacon (step 3), to its children. Because all ancestors have to shift their talk interval, node j also propagates the reverse beacon (step 4) up to its parent i at the n-th level. Note that in this case the schedule propagation impacts all nodes in the network. In fact, aside from the ancestors of the node which increases its talk interval, also other nodes can be involved in a transient phase which may require the introduction of a pause period. For instance, consider a node j' at the (n+1)-th level of the tree (illustrated in the second row of the scheme in the figure) which is not a direct ancestor of the node originating the new schedule. Its parent (i.e. node i) will shift ahead and advertise the new talk interval information during the m-th communication period. For reasons similar to the talk interval reduction, a pause period is introduced in the subtree rooted at nodes i, i.e. the nodes below the n-th level of the tree which are not direct ancestors of node k (which originated the new schedule). Assuming that (i) clocks of nodes are properly synchronized, and (ii) direct and reverse beacons never get lost, it can be shown that the following properties hold.

Property 1 (Schedule agreement). Child nodes wake up at the instant, and for the duration, enforced by their parent, even when talk intervals change.

Property 2 (Non overlapping schedules). For any two nodes i and j such that j is a child of i, the talk intervals i and j are not overlapped.

Property 3 (Adjacent schedules). In steady state conditions, the talk intervals shared by any node with its children and its parent, respectively, are contiguous.

C. *Schedule robustness Beacon Protection*

Beacon messages are critical for correctness of the protocol. When a node misses a direct beacon containing the new parameters, it cannot schedule its activity for the next communication period. In addition, the node cannot send direct beacons to its children until it re-acquires the correct schedule information. As a consequence, the loss of coordination propagates along the routing tree to its descendants. Direct beacons may get lost, for example, due to communication errors or collisions with other beacons or regular messages transmitted by interfering nodes. As direct beacons are sent through broadcast frames, they cannot be re-transmitted by the underlying MAC protocol. Instead, reverse beacons are unicast messages and, thus, they are retransmitted by the MAC protocol if not received correctly.

To add robustness to the direct beacon transmission and prevent collisions, the last part of the talk interval – referred to as *Beacon Period* – is reserved for the direct beacon transmission only. Child nodes must refrain from initiating regular message transmissions during the Beacon Period. In addition, the transmission of the direct beacon is initiated with a random back off delay. Finally, two back-to-back copies of the direct beacon are transmitted.

D. *Beacon Loss Compensation*

The Beacon Protection mechanism increases the probability that a direct beacon is successfully received by child nodes, but it does not solve the problem of direct-beacon losses. Therefore, we also devised the following mechanism to compensate the negative effects that derive from missing a direct beacon. Since talk intervals typically remain constant for a number of communication periods, when a node misses a direct beacon, it uses the current schedule parameters also in the next communication period. However, if the node misses the direct beacon even in the subsequent communication period, it remains awake until it re-acquires a new direct beacon.

Obviously, this heuristic produces a correct schedule if the parent node has not changed the talk interval in the meantime, which is true in almost all cases. Otherwise, it produces a non-optimal behaviour of the node (and its descendants as well) for a limited number of communication periods. The actual effect of a wrong prediction is different, depending on whether the talk interval has been increased or decreased. If a parent has reduced its talk interval, the corresponding child node wakes up earlier than the correct instant and we can now have overlapping schedules.

This results in energy wasting and useless message transmission that can potentially interfere with transmissions from other nodes. However, the child node remains awake until the end of the communication period and, very likely, receives a fresh direct beacon. On the other hand, if the talk interval has been increased, according to old schedule parameters, the child node wakes up at the right time but would go to sleep earlier than the correct instant.

However, since it missed the direct beacon in the previous communication period, it does not go to sleep until it receives a fresh direct beacon. Thus, it is very likely that it receives the new direct beacon almost immediately. If this is not the case, it will remain active until a new direct beacon is received.

III. LOAD BALANCING ALGORITHM

In a wsn, with a decentralized and heterogeneous structure, some nodes may have different capabilities of processing and batteries, imbalances of load can occur. Indeed, a more powerful node in term of processing capacity can become idle, because it has finished its work quickly while the others, less powerful, are occupied most of the time, consuming more energy. Powerful nodes capacity can be exploited by overloaded nodes if a fraction of their load is shared with them. If the difference between the heaviest loaded and the lightest loaded nodes is minimized, the average work execution time can be reduced, the energy of the nodes will be better exploited and the nodes lifetime can be extended. It is what contributes to the stability of the network topology that plays a principal role in different problems like: routing, scheduling, resource reservation etc. Load balancing is certainly one of the solutions for increasing the efficiency of applications and the network life time.

Load Balancing algorithms are designed essentially to distribute equally the load on nodes and maximize their utilization while minimizing the total task execution time.

This issue has been of considerable interest in the network research community when it comes to wired and wireless networks. It aims to guarantee that no node is under loaded or overloaded. It looks at setting up a uniform load on all nodes. Then, it is expanded in order to take into account new environments and new applications (large scale applications, multimedia applications, etc.). Compared to the wired networks, the mobile environments introduce new highly variable parameters such as limited resources, wireless link communication and mobility.

IV. IMPLEMENTATION AND EVALUATION

In this section, the implementation of the proposed schemes is given first. Then, the evaluated results on the proposed schemes are given.

A.*Implementation*

To evaluate the performance of ASLEEP, we implemented it by using the ns2 simulation tool.

B.Simulation setup

In both protocols of our analysis we referred to a network scenario consisting of 30-50 nodes randomly deployed over a 50x50 m2 area(as per existing authour's preferrence), with the sink placed at the center of the sensing area. Each node in the network generates a fixed number of messages per communication period, independent of its position on the routing tree. This scenario corresponds to a random deployment of sensor nodes over a given area for periodic reporting of sensed data, which is a typical case in monitoring applications. In such applications, data of interest (e.g., temperature, vibrations) are sensed and reported periodically to the sink node – data messages are typically short e.g., 10-20 bytes [11]. The sensing/communication period depends on the specific application.

TABLE 1. SIMULATION PARAMETERS

Parameter	Value	Parameter	Value
Communication Period (CP)	30 s	Average Error-Burst Size	5.7 ms
Message rate	1 msg/CP	Average Error-free Burst Size	46.2 ms
Message size MAC frame size	20 bytes 40 bytes	Observation window (L) TI time slot (q)	10 CPs 100 ms
Transmission Range	15 m	Beacon Period	60 ms
Carrier Sensing Range Message Error/Loss Rate	30 m 10%	TI decrease time threshold (L_{down}) TI decrease threshold (g_{down})	5 CPs 2q (200 ms)

C.Analysis in dynamic conditions

To investigate the behaviour of ASLEEP in dynamic conditions we considered two different kinds of variation in the operating conditions.

• *Traffic pattern variation*. Sensor nodes start with a given message generation rate. Then, after some time, they increase significantly their message rate and, finally, they switch back to the original rate. This scenario may occur when sensors are requested to report an event with better fidelity (i.e., using a higher sampling rate or including additional physical quantities) for a limited time.

• *Topology variation*. These experiments start with an initial configuration where only one half of the nodes deployed in the sensing area report data. After some time, also the remaining nodes start reporting data. This scenario may occur when additional nodes are required to report data so as to observe the sensed phenomenon with increased spatial resolution.

The following performance indices are measured:

• Talk interval. Plotting the talk interval duration over time provides a graphical representation of the ability of the protocol to adapt to changing operating conditions.

• Duty-cycle, denotes the fraction of time a sensor node is active within a communication period .

• Transient time duration, defined as the number of communication periods from when the variation occurs to when the new talk interval stabilizes. We considered a talk interval as stable when it remains constant for more than L communication periods. This metric gives a measure of how quickly the protocol adapts to the new operating conditions.

V .RESULTS

Since the analysis in dynamic conditions is aimed at investigating how the protocol reacts to changes in the traffic pattern and network topology, we did not consider the effects of message errors/losses in this part. This allow us to better understand the behavior of the protocol.

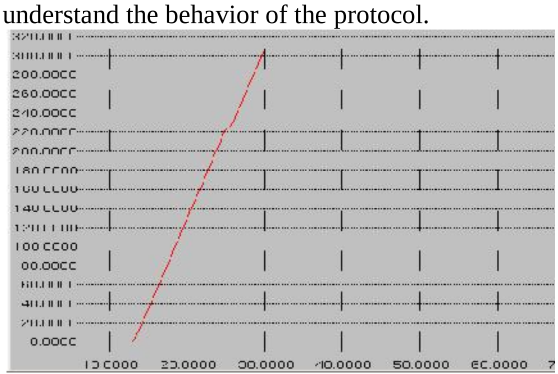

Figure 3 Energy of load node(ENERGY vs TIME)

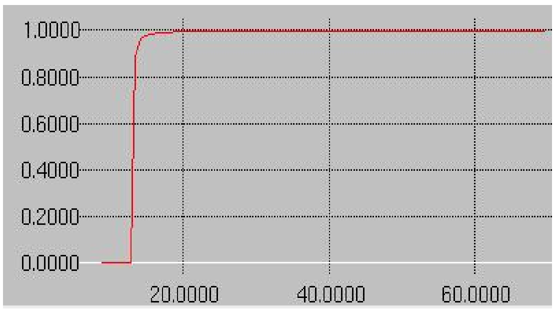

Figure 4 PDR of Alterate node (PDR vs TIME)

REFERENCES

[1] Adreas Merentitis, Nektarios Kranitis, Antonis Paschalis, and Dimitris Gizopoulos," Low Energy Online Self-Test of Embedded Processors in Dependable WSN Nodes"(2012) IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING,VOL. 9, NO. 1

[2] Akyildiz I.F., Su W., Sankarasubramaniam Y., and E. Cayirci,(2002) "Wireless Sensor Networks: a Survey", Computer Networks, Vol.38, No: 4.

[3] Anastasi G., Conti M., Castronuovo M., Di Francesco M., "Experimental Evaluation of an Adaptive Staggered Sleep Protocol for Wireless Sensor Networks"(2008), Proc. IEEE WoWMoM, International Workshop on Experimental Activities in Wireless Networks and Systems

[4] Anastasi G.,Conti M., Di Francesco M., "An Adaptive Sleep Strategy for Energy Conservation in Wireless Sensor Networks"(2009),

Technical Report DII-TR http://info.iet.unipi.it/~anastasi/papers/DII-TR-2009-03.pdf

[5] Chaudhary D.D., Pranav Pawar and Waghmare L.M.,“Comparison and Performance Evaluation of Wireless Sensor Network with different Routing Protocols” (2011) ,IPCSIT vol.6

[6] FENGYUAN REN, JIAO ZHANG, TAO HE, CHUANG LIN, and DAS S.K.,(2011),“Energy-Balanced Routing Protocol for Data Gathering in Wireless Sensor Networks” IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 22, NO. 12,

[7] Ganesan D., Cerpa A., Ye, Yu Y., Zhao J., Estrin D., (2004) “Networking Issues in Wireless Sensor Networks”, Journal of Parallel and Distributed Computing, Vol. 64, pp. 799-814.

[8] Giuseppe Anastasi , Marco Conti, Mario Di Francesco: “Extending the Lifetime of Wireless Sensor Networks through Adaptive Sleep”(2010), - IEEE Transactions on Industrustrial Informatics.

[9] ISMAIL AHMEDY, ASRI NGADI, SYARIL NIZAM OMAR and JUNAID CHAUDHRY (2011) “A review on wireless sensor networks routing protocol: Challenge in energy perspective” Scientific Research and Essays Vol. 6(26).

[10] Jong-Shin Chen, Neng-Chung Wang , San-Heui Jhuang , (2010) “Efficient Cluster Head Selection Methods for Wireless Sensor Networks” JOURNAL OF NETWORKS, VOL. 5, NO. 8

[11] Kansal A., Hsu J., Zahedi S., Srivastava M., ”,(2007) “Power Management in Energy Harvesting Sensor Networks,ACM Transactions on Embedded Computing Systems, Vol. 6, N. 4,.

[12] Kenan Xu, Hossam Hassanein, Glen Takahara, and Quanhong Wang, (2010) ” Relay Node Deployment Strategies in Heterogeneous Wireless Sensor Networks” -IEEE TRANSACTIONS ON MOBILE COMPUTING, VOL. 9, NO. 2

[13] Kobi Cohen and Amir Leshem, (2010) “A Time-Varying Opportunistic Approach to Lifetime Maximization of Wireless Sensor Networks”, IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 58, NO. 10

[14] Mirza D., Owrang M., Schurgers C., (2005) “Energy-efficient Wakeup Scheduling for Maximizing Lifetime of IEEE 802.15.4 Networks", Proc. Int’l Conference on Wireless Internet , Budapest - Hungary.

[15] Mohamed K. Watfa, Haitham AlHassanieh, and Samir Selman, (2011) “Multi-Hop Wireless Energy Transfer in WSNs”,IEEE COMMUNICATIONS LETTERS, VOL. 15,NO. 12

[16] Wanqing Tu, (2012)” Efficient Resource Utilization for Multi-Flow Wireless Multicasting Transmissions”, - IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 30, NO. 7

[17] Wei Ye,,and Fernando,(2008)“Robust Optimization Models for Energy-Limited-Wireless Sensor Networks”-IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, VOL. 7, NO. 6

Arun Thangavel pursing Master of Engineering in Applied Electronics at K.S.R College of Engineering. He presented a paper in an International conference at Karunya University. His research interests include wireless sensor network, network lifetime and energy.

B.Yuvarani received B.E degree in Electrical and Electronics Engineering in VMKV in 2001. She received M.E degree in Power Electronics and Drives in Sona College of Technology in 2010. Her area of interest includes Power Electronics and Drives.