Adaptive Media Access Control for Energy Harvesting - Wireless Sensor Networks

Xenofon Fafoutis and Nicola Dragoni
Technical University of Denmark, DTU Informatics, Denmark
{xefa,ndra}@imm.dtu.dk

***Abstract*—ODMAC (On-Demand Media Access Control) is a recently proposed MAC protocol designed to support individual duty cycles for Energy Harvesting - Wireless Sensor Networks (EH-WSNs). Individual duty cycles are vital for EH-WSNs, because they allow nodes to adapt their energy consumption to the ever-changing environmental energy sources. In this paper, we present an improved and extended version of ODMAC and we analyze it by means of an analytical model that can approximate several performance metrics in an arbitrary network topology. The simulations and the analytical experiments show ODMAC's ability to satisfy three key properties of EH-WSNs: adaptability of energy consumption, distributed energy-aware load balancing and support for different application-specific requirements.**

***Index Terms*—Wireless sensor networks, Multiaccess communication**

I. INTRODUCTION

Energy Harvesting - Wireless Sensor Networks (EH-WSNs) [6] consist of nodes that are capable to extract energy from the environment. Each node may also be equipped with an energy buffer. If the harvested energy is more than the consumed energy, then the node operates at a sustainable state and has a continuous lifetime. This state is defined as *Energy Neutral Operation (ENO)*. Operating states where the harvested energy is much higher than the consumed energy are sustainable yet suboptimal, as the excess of energy is wasted. The desired operating state is when the harvested energy is equal to the consumed energy, since it is a sustainable state where all the harvested energy is used to improve the system performance. Operating at this state, which is named *ENO-Max* [9], constitutes a foundational goal of EH-WSNs.

Aiming at ENO-Max, communication protocols for EH-WSNs need to be able to adapt their energy consumption to the availability of environmental energy, which varies over space and time. Furthermore, communication protocols need to have certain additional qualities. First, they need to distribute the load to the nodes that have access to more energy at any given time. Secondly, they need to support applications with different performance requirements. In other words, the harvested energy should be used to improve the application performance.

In this paper, we present an extended version of ODMAC (On-Demand Media Access Control) [3] and we study it using an analytical model that can approximate several performance metrics in arbitrary network topologies. The analytical experiments show ODMAC's ability to provide the aforementioned qualities: adaptability of energy consumption, distributed energy-aware load balancing and support for different application-specific requirements. Previous simulations in OPNET verify the same findings [3].

The paper is organized as follows. The new ODMAC is presented in Sec. II. In Sec. III we introduce a model of an arbitrary EH-WSN that uses ODMAC. This allows us to analytically study the protocol in Sec. IV. Finally, Sec. V discusses the related work and VI concludes the paper.

II. ON-DEMAND MAC (ODMAC)

ODMAC uses the carrier sensing scheme in order to support individual duty cycles. Each receiver periodically broadcasts a beacon which indicates to its receivers that it is ready to accept incoming data packet transmissions. All nodes that have queued packets that need to be forwarded to the sink, are listening to the channel waiting for an appropriate beacon. Upon receiving the beacon, the data packet transmission follows. To decrease the beacon waiting time the protocol incorporates an *opportunistic forwarding scheme*. Instead of waiting for a specific beacon, the transmitter opportunistically forwards each frame to the sender of first beacon received as long as it is included in a list of appropriate forwarders. Apart from the reduction of the waiting time, this approach allows the nodes to control the packet-relaying load distribution between them. In addition to the opportunistic mode, in this paper we introduce a *binding scheme* which is more suitable in some exceptional cases. In this forwarding scheme, the transmitter selects one node and binds to its individual duty cycle. This approach decreases the energy consumption as it completely eliminates idle listening at the cost of additional delays. Binding is automatically used when there is only a single appropriate next hop. It can be also used in extreme low power conditions for additional energy savings.

Fig. 1 depicts the basic communication between an ODMAC transmitter and an ODMAC receiver. Assume that the sensor node B needs to forward one packet to A. Node B listens the channel waiting for a beacon. At some point, node A wakes up and attempts to transmit a beacon.

This work was partially supported by the IDEA4CPS project granted by the the Danish Research Foundation for Basic Research.

978-1-4673-1786-3/12/$31.00

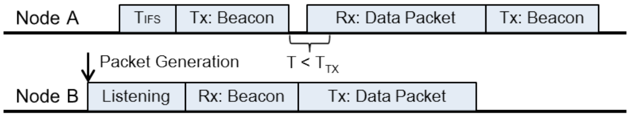

Fig. 1. Communication between a transmitter and a receiver.

First, node A listens to the channel for an amount of time (T_{IFS}). Unless the channel is free throughout all that time, node A returns to a sleeping state. This reassures that the communication between two nodes remains uninterrupted. If the channel is free, node A transmits the beacon and initiates a waiting timer, T_{TX}, while waiting for incoming packets. If no node transmits during the defined time, node A returns to the sleeping state. Node B receives the beacon, identifies that the beacon originates from node A and transmits the packet. Lastly, we opt for removing the collision protection mechanism. The reason is that the beaconing period needs to be several times shorter than the sensing period, as shown in the following sections. Hence, the stochastic selection of a beacon to transmit constitutes an effective indirect collision avoidance mechanism.

III. MODELING ARBITRARY ODMAC EH-WSNS

Waiting-for-a-Beacon Delay. Suppose that each one of the forwarding candidates has a beacon period, t_j. Let X_j be the waiting time for the beacon of node j and x_j its expected value. The node forwards the frame to the node that wakes up first. Let Y_i be the waiting time for the first appropriate beacon out of the n potential next hops and y_i its expected value. Hence, $P(Y_i \leq y_i) = 0.5$. Given that X_J follows a uniform distribution, $P(X_j > y_i) = (t_j - y_i)/t_j$, Eq. (1) gives us the expected waiting time for the first beacon (y_i).

$$P(Y_i \leq y_i) = 1 - \prod_{j=1}^{n} P(X_j > y_i) = 1 - \prod_{j=1}^{n} \frac{t_j - y_i}{t_j} \quad (1)$$

Node-to-Sink Delay. The node-to-sink delay is composed by the sum of every link delay in each intermediate hop. We consider significant only the transmission delay and the waiting delay signficant sources of delay in each link. The transmission delay is equal to L/R, where L is the packet size and R is the transmission rate of the link. The waiting delay is given by (1). The sum of those gives us the link delay for node i, $d_i^l = L/R + y_i$. Remember that paths are opportunistically decided. Each one of the appropriate forwarders is expected to serve a portion of the the node's packets. The probability the packet will be forwarded by node i is given by p_i where the sum iterates over the nodes that are in the list of appropriate forwarders.

$$p_i = \frac{1}{t_i \sum\limits_{j=0}^{n} \frac{1}{t_j}} \quad (2)$$

The node-to-sink delay (d_i^s) in sensor i is equal to the local link delay (d_i^l) plus the respective node-to-sink delay of each potential forwarder with respect to the probability of it being the actual forwarder. This is given by the following equation where the sum iterates over the nodes that are in the list of appropriate forwarders.

$$d_i^s = d_i^l + \sum_{j=0}^{n} p_j d_j^e \quad \text{[sec]} \quad (3)$$

For the nodes that have direct access to the sink, Eq. (3) still applies with $p_{sink} = 1$ and $d_{sink}^e = L/R$.

Traffic Rate. The traffic a sensor needs to transmit (r_i) consists of the traffic it generates by sensing (r_i^g) and the traffic it forwards on behalf of other nodes (r_i^f). The traffic rate generated locally is equal to $r_i^g = 1/s_i$, where s_i is the period of the sensing duty cycle. In addition to that, every backwards neighbor contributes with a part of its total traffic rate with respect to the probability of node i being the actual forwarder (given by (2)). The latter is given by the following equation where the sum iterates over the nodes that have node i in their list of appropriate forwarders.

$$r_i = \frac{1}{s_i} + \sum_{k=0}^{m} p_k r_k \quad \text{[1/sec]} \quad (4)$$

For the nodes that are in the outer layer of the network, Eq. (4) still applies with $m = 0$.

Power Consumption and Generation. We model only the power consumed in communication, as it is the most significant source of power consumption. The total power consumed for transmitting packets (P_i^{ttx}) is given by (5) where r_i is given by (4), the ratio of the packet size (L) over the transmission rate (R) is the duration of the transmission and P_i^t is the power consumed while transmitting.

$$P_i^{ttx} = P_i^t r_i \frac{L}{R} \quad \text{[W]} \quad (5)$$

For the value of P_i^t, we use the power consumption model presented in [10]. In particular, the power consumed in transmission is given by the following formula where P_i^{tx} is the selected power of the transmitted signal, η is the drain efficiency and P_{t0} is the power consumed in the circuits of the communication module constantly.

$$P_i^t = P_{t0} + \frac{P_i^{tx}}{\eta} \quad \text{[W]} \quad (6)$$

The total power consumed for receiving packets (P_i^{trx}) is given by the following formula where P_r is the power consumed in reception, r_i^f is the traffic rate of the forwarded packets and the ratio of the packet size (L) over the transmission rate (R) is the time required for the reception.

$$P_i^{trx} = P_r r_i^f \frac{L}{R} \quad \text{[W]} \quad (7)$$

The total power consumed while waiting for an appropriate beacon (P_i^w) depends on weather the node operates in opportunistic or binding mode. In the latter case, no power is consumed, $P_i^w = 0$. In the former case, the power consumed is given by the following formula where P_r is the power consumed in reception, y_i is the waiting time given by (1) and r_i is given by (4).

$$P_i^w = P_r y_i r_i \quad \text{[W]} \tag{8}$$

Lastly, the total power consumed for beaconing (P_i^b) is given by the following formula where t_i is the beaconing period, the ratio of the beacon size (L_b) over the transmission rate (R) is the time required for a beacon transmission and P_t is the power consumed while transmitting.

$$P_i^b = P_i^t \frac{1}{t_i} \frac{L_b}{R} \quad \text{[W]} \tag{9}$$

We choose not to model potential collisions, as we consider them insignificant in the given light traffic conditions. The sum of all the aforementioned sources of energy consumption give the total power consumption (P_i^{tot}) of node i. The harvested energy, P_i^{in}, is modeled as a random variable that follows a normal distribution. Whenever $P_i^{in} \geq P_i^{tot}$, node i operates at an ENO state. If the ratio is in $[1, 1.1]$, we consider the node to operate in ENO-Max state.

Transmission Range. The transmission range model is based on the link budget formula. P_i^{rx} is signal's power at the receiver in dBm, P_i^{tx} is the power of the transmitted signal in dBm, G^{tx} and G^{rx} are the antenna gains at the transmitter and receiver in dBi, respectively, and PL_i is the signal attenuation over the path, i.e. path loss, in dB. We consider the antenna gains to be the same at all nodes.

$$P_i^{rx} = P_i^{tx} + G^{tx} + G^{rx} - PL_i \quad \text{[dBm]} \tag{10}$$

The path loss at a distance d_i is given by the following equation assuming free space model , where f is the frequency of the signal (MHz) and e is the loss exponent.

$$PL_i = 20\log(f) + 10\log(d_i^e) - 27.55 \quad \text{[dBm]} \tag{11}$$

If we equate P_i^{rx} to the receiver's sensitivity threshold, we get the transmission range, d_i, of node i.

Arbitrary Topologies. The presented formulae effectively model an EH-WSN. Given an arbitrary set of nodes, with either predefined or random positions in $A \times A$ field, and a set of input parameters for each one of them, we can find the topology and approximate the performance metrics.

IV. ANALYSIS OF ODMAC

Table I provides the values of some parameters of the model. These values apply to all sensor nodes. The parameters suppose using the CC1000 transceivers [10].

TABLE I
VALUES OF MODEL PARAMETERS

L	100 Bytes	G	0 dBi	P^{tx}	10 dBm
L_b	8 Bytes	e	4	η	0.157
R	256 Kbps	P^{rx}	−96 dBm	P^{t0}	15.9 mW
f	433 MHz	A	300 m	P^{r0}	22.2 mW

Additionally, we select the maximum supported transmission power and then we gradually decrease it to the point that no links are broken. The list of appropriate forwarders includes all the nodes that are one hop closer to the sink. Lastly, we consider a random topology of 50 nodes. Based on these parameters, the transmission range is approximately 105 meters. The sink node is placed in position $(0, 0)$, leading to a 5-hop deep network. Further experiments in different random topologies verify the same trends. However, they are omitted due to space limitations.

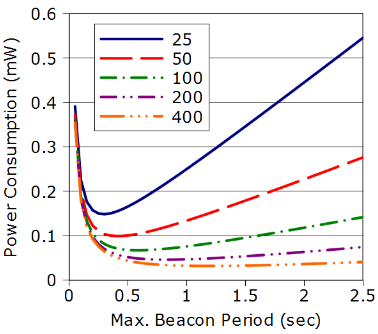

Fig. 2. Power consumption for different max. sensing periods (s_{max})

Analysis of Beacon (t_i) and Sensing Period (s_i). Increasing the beacon period (t_i) has two opposite effects in the power consumption. From one side, the power consumption due to beaconing is decreased. On the other side, the nodes that depend on the node's beacon need to spend more time waiting for a beacon, wasting energy in idle listening. In this experiment, we consider the worst case scenario when the energy input is so low, that all nodes operate at the maximum duty cycles periods, t_{max} and s_{max} respectively. Fig. 2 shows the average power consumption in each node for different maximum beaconing periods. Different lines represent different maximum sensing periods in seconds. Observe the minimum that gradually increases as the sensing period increases. We set the maximum beaconing period t_{max} to the value of that minimum. Thus, the system has the following operating alternatives. The system can trade power for shorter delays if t_i is adapted in $(0, t_{max}]$. The system can also trade power for throughput by adapting s_i. Alternative, the system can operate at the minimum power consumption and use the energy elsewhere (e.g. security).

Case Study: Delay-Sensitive Applications. Aiming to support delay-sensitive applications, the system should reassure energy neutral operation and invest the excess of harvested energy in decreasing the delays. We assume that the applications are characterized by a maximum sensing period requirement in seconds, s_{max}. Similarly, a minimum sensing period is defined, s_{min}. A duty cycle adaptation algorithm is defined as follows. All sensors set their sensing period to $s_i = s_{min}$ and their beaconing period to $t_i = t_{max}$, where t_{max} is given by Fig. 2. If a node has an excess of energy, it decreases the beaconing period. If, on the other hand, a node needs to save energy, it first increases the beaconing period up to the maximum value, t_{max}. If this is not enough to achieve a ENO state, the sensing period is increased up to its maximum value, s_{max}. If this is still not enough, the node switches to *binding mode* and it binds to the node with the minimum beaconing period.

Table II shows the results of several numerical experiments. The EH-WSN is tested under four energy input conditions (*mean / variance*) and under different application requirements (s_{max}). The energy inputs cover a large variety of energy harvesters according to [6]. We also consider that $s_{min} = s_{max}/2$. The table shows the average sensing rate of the nodes in packets per minute and the average node-to-sink delay in ms. The last column gives the overall energy state of the system where *ENO-Max* means that all the nodes operate in *ENO-Max*. Otherwise, the number of nodes that operate at an unsustainable state is given.

TABLE II
NUMERICAL RESULTS FOR DELAY-SENSITIVE APPLICATIONS

$P^{in}(mW)$	$s_{max}(s)$	rate (ppm)	delay (ms)	state
1 / 0.2	50	2.4	21.2	ENO-Max
1 / 0.2	100	1.2	23.1	ENO-Max
1 / 0.2	200	0.6	21.5	ENO-Max
0.3 / 0.05	50	2.36	48.7	ENO-Max
0.3 / 0.05	100	1.2	57.1	ENO-Max
0.3 / 0.05	200	0.6	48.4	ENO-Max
0.1 / 0.02	50	1.57	238.1	3
0.1 / 0.02	100	0.94	215.1	1
0.1 / 0.02	200	0.54	187	ENO-Max
0.01 / 0.002	3200	0.019	1600	ENO-Max

Under high power input, the system operates at the maximum desired sensing rate while the excess of energy is used decrease the node-to-sink delay as much as possible. All nodes operate at *ENO-Max*. As we decrease the energy input, the average node-to-sink delay gets higher, which shows that the system effectively uses the harvested energy to improve the selected performance metric. When the power input is even lower, many nodes need to switch to *binding mode* to achieve an *ENO* state and the system achieves *ENO-Max* when we loose the application sensing rate requirements. In the last case, the network manages to operate at a sustainable state in very low power conditions by generating approximately one packet per hour.

To summarize the key conclusions of this experiment, ODMAC can effectively adapt its energy consumption to different energy inputs of various orders of magnitude, providing a sustainable operation. Additionally, we see that that the delays are decreased as the system it exposed to higher levels of energy. This shows that the harvested energy is used to favor the performance metric that is selected to be the most important. Further experiments on applications with different requirements, that are omitted due to space limitations, verify these findings.

V. RELATED WORK

The authors of [2] studied several MAC layer approaches for EH-WSNs. An important limitation of their work is that they consider only the case of single-hop WSNs. This is a simplified case, as the sink does not have energy constraints and node synchronization is not a challenge. Thus, those schemes cannot be directly applied in multi-hop scenarios.

Duty cycling introduces a challenge in node synchronization as a transmitter does not trivially know if the receiver is awake. Traditionally in WSNs, there are several approaches to this problem. In synchronization approaches, such as S-MAC [11] and T-MAC [8] the nodes form virtual clusters that share a common sleeping schedule. This approach is unsuitable for EH-WSNs, as it cannot support *individual* duty cycles that are essential for adapting the energy consumption to the ambient energy that is available to each individual node. In preamble approaches, such as B-MAC [5] and X-MAC [1], the senders transmit a preamble lasting at least as long as the sleeping period of the receiver before the actual data transmission. When the receiver wakes up and detects the preamble, it stays awake for the data transmission. In beacon baseing approaches, such as RI-MAC [7] and ODMAC, the communication is itinitated by the receiver through beacons. We consider the latter approach the most suitable for our domain. A comparison of the two asynchronous approaches is provided in [4].

VI. CONCLUSIONS

Energy sources available for harvesting are characterized by a significant variation over space and time. This constitutes decoupled duty cycles essential. In this paper we have presented an extended version of ODMAC. We have shown that using ODMAC, the nodes are able to adjust their operation to sustainable levels in various energy conditions that cover a vast amount of energy harvesting technologies. Furthermore, our results show that the energy can be used to favor different application requirements.

REFERENCES

[1] M. Buettner, G. V. Yee, E. Anderson, and R. Han. X-mac: A short preamble mac protocol for duty-cycled wireless sensor networks. In *SenSys*, 2006.

[2] Z. A. Eu, H.-P. Tan, and W. K. Seah. Design and performance analysis of mac schemes for wireless sensor networks powered by ambient energy harvesting. *Ad Hoc Networks*, 9, 2011.

[3] X. Fafoutis and N. Dragoni. ODMAC: An On-Demand MAC Protocol for Energy Harvesting - Wireless Sensor Networks. In *PE-WASUN*, 2011.

[4] X. Fafoutis and N. Dragoni. Analytical Comparison of MAC Schemes for Energy Harvesting - Wireless Sensor Networks. In *EnHaNSS*, 2012.

[5] J. Polastre, J. Hill, and D. Culler. Versatile Low Power Media Access for Wireless Sensor Networks. In *SenSys*, 2004.

[6] W. K. G. Seah, Z. A. Eu, and H.-p. Tan. Wireless Sensor Networks Powered by Ambient Energy Harvesting (WSN-HEAP): Survey and Challenges. In *Wireless VITAE*, pages 1–5, 2009.

[7] Y. Sun, O. Gurewitz, and D. B. Johnson. RI-MAC: A Receiver-Initiated Asynchronous Duty Cycle MAC Protocol for Dynamic Traffic Loads in Wireless Sensor Networks. In *SenSys*, 2008.

[8] T. van Dam and K. Langendoen. An adaptive energy-efficient mac protocol for wireless sensor networks. In *SenSys*, 2003.

[9] C. M. Vigorito, D. Ganesan, and A. G. Barto. A.g.: Adaptive control of duty cycling in energy-harvesting wireless sensor networks. In *IEEE SECON*, pages 21–30, 2007.

[10] Q. Wang, M. Hempstead, and W. Yang. A Realistic Power Consumption Model for Wireless Sensor Network Devices. In *IEEE SECON*, pages 286–295, 2006.

[11] W. Ye, J. Heidemann, and D. Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Trans. Netw.*, 12:493–506, 2004.