Increasing energy efficiency in WSNs using wakeup signal length optimization combined with payload aggregation and FEC

Ákos Milánkovich[1], Gergely Ill[1], Károly Lendvai[1], Sándor Imre[1] and Sándor Szabó[1]

***Abstract*—Energy efficiency in wireless sensor networks is a vital question. There are several possibilities to achieve longer battery life in such devices. We investigated delay-tolerant wireless sensor networks with battery-operated nodes and use data-aggregation to lower the size of transmitted data overhead caused by packet headers.**
In this paper a mathematical formula is presented to calculate the optimal wakeup signal (a special radio signal) length, which minimizes the energy consumed for waking up nodes in sleep mode. The demonstrated results and graphs are based on the investigation of an existing system. The contribution of this paper is a general method to improve the energy efficiency of wireless sensor networks by using the optimal length of the wakeup signal in case of different amounts of aggregated packet payloads and Forward Error Correction (FEC) schemes. The results presented can be applied to arbitrary packet-based wireless protocols and radio modules supporting wakeup signal listening.

***Keywords*—energy efficiency, wireless sensor networks, aggregation, sleep-wake cycle, wakeup signal length, WSN, FEC, DTN**

I. INTRODUCTION

Wireless technologies drive the innovation in the telecommunication sector [1]. One of the key areas, wireless sensor networks is becoming popular in various scenarios such as environment, production and health care monitoring, intelligent home, precision agriculture, smart metering, etc. In the design and implementation phase of these systems, special attention should be paid to the energy consumption of the network nodes, since these in many cases operate on battery power. Moreover, in case of Delay-Tolerant Networks (DTN), it is possible that the nodes transmit the useful information in an application-specific predefined time T delay instead of real-time communication.

This paper focuses on the energy consumption of sensor networks with the restrictions defined by the operation of DTNs. Our goal is to minimize the energy consumption of network nodes, taking into account the BER (Bit Error Ratio) quality of the radio channel to maximize battery life. This paper aims to reach this goal by finding the optimal length of the wakeup signal. The method was developed for multi-hop wireless sensor networks with stationary nodes.

[1]Department of Networked Systems and Services
Budapest University of Technology and Economics
Budapest, Hungary
Email: {milankovich, ill, lendvai, imre, szabos}@hit.bme.hu
Manuscript received: August 14, 2015. Revised: November 23, 2015

This paper is the extended version of [2]. In this paper solutions for sleep-wake optimization is presented. The problem is gaining attention nowadays as the popularity of sensor networks is rapidly increasing. Besides the solutions presented here, there are other approaches, but the basic idea behind them is similar to the discussed protocols. Among the published solutions there are synchronous and asynchronous scheduling methods and also low-energy MAC (Medium Access Control) protocols. The WSN (Wireless Sensor Network) community often refers to this problem as "low power listening".

This paper is organized as follows: Following the related work section, Section III. introduces the system model along with the considered parameters of the sensor network hardware and communication protocol. Section IV. shows the benefits of a better sleep-wake scheduling. Next, in Section V. the formulas for optimization and the results are presented. Finally, Section VI. concludes the paper.

II. RELATED WORK

The solution [3] employs relay nodes. These intermediate nodes can be installed easily and can be used to increase the reliability of the communication network. Moreover they can increase the energy efficiency of the network. In this solution the task is to select the relay node(s) to achieve the most energy-efficient routing. To optimize between energy efficiency and load balancing, authors determine the amount of energy required for the transmission and reception with proper QoS. Therefore the system can chose from multiple relays and energy levels combined with variable transmission power and cooperative sleep-wake scheduling.

S-MAC [4] handles the problem in the Medium Access Control layer. According to the protocol all nodes can be in one of these states: sleep, awake and listen. In sleep mode, the nodes turn their radios off and set a timer to wake up later. The length of listening and sleeping periods can be tuned for the applied scenario. The neighbor nodes are synchronized, so that they fall asleep and wake up at the same time. The nodes share their sleep-wake schedules with their neighbors, and store it in a table. Moreover, they can communicate with each other without perfect synchronization. Medium access is achieved by using RTS/CTS mechanism. Beside the advantages of S-MAC, there are also some disadvantages. The hop-by-hop delay may increase, which can be a problem in some applications. Also, every node has to maintain a scheduling table, which can consume a significant amount of memory in case of many neighbors. To handle the scheduling the microcontroller has to stay awake continuously.

In case of the third method [5] the network consists of sensor nodes and sinks (data collectors). The sensor nodes are responsible for detecting events and sending packets to the sinks via multi-hop. The sinks are connected with wired links and have infinite power sources. For energy efficiency purposes, the nodes use asynchronous sleep-wake scheduling. The waking events are considered to be Poisson processes with parameter λ. Therefore the wake intensity means the frequency of switching to active state, and influences the energy spared during sleep state and also the bandwidth. The authors defined an overhead measure, which determines the amount of energy needed beyond the data transfer. The goal is to minimize this overhead by changing some variables, providing sufficient bandwidth according to the nodes data generation intensity, and to achieve, that in average a certain percent of nodes from the forwarding set should be awake to forward the data.

Another class of papers [6] [7] [8] [9] introduce a different approach to wakeup listening. They propose the use of an additional low-power radio module, which has the responsibility of receiving wakeup packets (in most cases out-of-band, and in rare cases they are even capable of addressing) and notify the microcontroller to switch on the main radio for the reception of the real packet. These papers suggest that the use of an additional low-power radio could significantly reduce the overall energy consumption compared to continuous idle listening. These papers sacrifice the radio range to achieve lower power. In this paper the authors preserve the radio range of the original radio module. Moreover the circuitry of the module is simpler, thus cheaper using only one radio module.

III. SYSTEM MODEL

A. Communication Protocol

In this section the operation of a communication protocol is presented as an example, which will be used in the formal mathematical model to show results. In the example communication protocol the header and trailer both have fixed length determined by the applied communication protocol, the types of encryption and error correction code. From the point of transmitted useful data, these are overhead. The combined length of the header and trailer is ω bits. The useful data consists of fix, predetermined length of elements and structure. The size of this payload data is φ bits. To maximize the energy efficiency of the system, the useful bits/all transmitted bits ratio has to be maximized. Assuming no error in the transmission the most possible useful data can be transmitted in one packet, which means, that aggregating the information into one packet is necessary, and guarantees that the overhead ratio in the packet is minimal. In a data packet, n pieces of data elements of φ bits length are transmitted, so the useful data amount is n times φ bits. Figure 1 shows the communication flow between a sender and a receiver node. The sender indicates its intention of sending a packet to the receiver node by broadcasting a wakeup signal, containing a special (longer) preamble which can be recognized by the RF chip. The application of such special preamble has to be supported by the RF chip hardware.

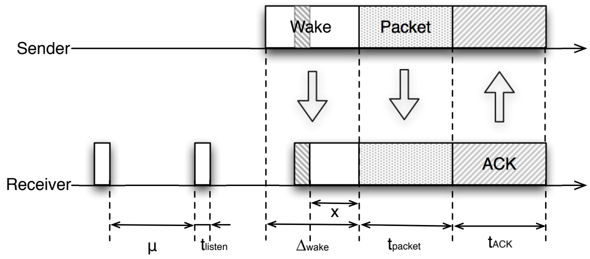

Fig. 1: Communication flow between two nodes

Some vendors refer to this functionality as WOR (Wake on Radio). Texas Instruments CC1101 [10] used in this paper supports this feature. The wake message contains the node ID of the destination node as well. Immediately after the wake message, the sender sends the packet, and then waits for acknowledgment. In case the ACK did not arrive in time, the packet is considered to be lost and will be resent later. Meanwhile the receiver nodes are listening for wakeup signals with μ periodicity. To successfully receive a wakeup signal, nodes need to listen for at least time t_{listen}. If the wakeup was successful, the receiver listens for the data packet. Otherwise the node was not awakened and the transmission was not successful. After the packet was successfully received, the receiver sends an ACK to the sender node.

In addition to the communication flow described above, some additional assumptions were made:

- The nodes in sensor networks usually have more states: sleeping, receiving and sending. In sleep mode the nodes turn their radio modules off, and set a timer to wake up later.
- The duration of signal propagation on the radio channel is considered to be zero,
- The wakeup signal always successfully wakes up the nodes,
- The radio channel is symmetrical for BER and PER,
- The packets never collide with other packets on the radio channel (to make the modeling easier),
- A node always receives one packet at once,
- The storage memory of the nodes is infinite, without restriction for packet length.

B. Model parameters

In this section, we introduce the parameters used in the following formulas. The parameters and their values are summarized in Table I. The demo system consists of an Atmel AVR XMEGA A3 microcontroller [11] and a TI CC1101 433 MHz radio module[10]. Both devices are extremely suitable for sensor networks, due to their low power consumption, reliability and low price.

B : 9.6 kbaud/sec. Using GFSK modulation, one symbol carries one bit, which equals 9.6 kbit/sec.

i_{tx} : 40 mA (at +10 dBm output power). This value should

TABLE I: Parameters for calculating optimal wake time

Symbol	Description	Value	Unit
ω_h	length of header	128	bit
ω_{MAC}	length of MAC	16	bit
B	transfer rate	9600	bit/s
n	aggregation number	1-100	pcs
φ	length of payload	80	bit
BER	bit error rate	4E-3,4E-4,4E-5	prob.
N	block size of FEC	depends on FEC	bit
K	code length of FEC	depends on FEC	bit
t	error correcting capability of FEC	depends on FEC	bit
r	number of retransmissions	depends on FEC and BER	pcs
i_{rx}	RX current	20	mA
i_{tx}	TX current	40	mA
i_{idle}	Current in SLEEP mode	0.031	mA
σ	Packet size	Depends on n	bit
$t_{WaitForACK}$	Expected waiting time for ACK (including processing and guard times)	1	s
t_{listen}	Listening time for successful awaking	0.073	s
T	Examined period length	1	h
λ_t	Number of sent packets during T	1-60	pcs
λ_r	Number of received packets during T	1-60	pcs
C_{src}	Battery stored energy	8500	mAh

be increased by the 1340 μA current draw of the microcontroller, but in case of transmission, the microcontroller encodes simultaneously, so this value is considered in I_{enc}. ([10] page 9, Table 4.)

i_{rx} : 20 mA (at sensitivity limit). This value should be increased by the 1340 μA current draw of the microcontroller, but similarly as the transmission, in case of receiving, the microcontroller simultaneously decodes, so this value is considered in I_{dec}. ([10] page 10, Table 4.)

C. Forward Error Correction schemes

In this article the authors use block codes for error correction, because their implementation requires fewer resources – from the limited computational capacity of microcontrollers - than other more advanced codes. The following three error correction codes were considered:

Hamming codes [12] are basic linear block codes [13] using parity checking as the added redundant information. They can only correct one bit per block and detect 2 incorrect bits. Hamming codes are perfect codes [13] and can be decoded using syndrome decoding [14]. They are often used in ECC memory modules.

Reed-Solomon [15], [16] codes are cyclic BCH codes. They are commonly used in CDs and DVDs.

BCH (Bose-Chaudhuri-Hocquenghem) [17] codes are also linear block codes, which can be defined by a generator polynomial.

To calculate the energy consumption of a Forward Error Correction (FEC) scheme, first the execution time of every FEC scheme on the same computer using Matlab simulation was measured. We chose this platform, as most of the FEC codes are already built-in. Then we implemented the selected code of each FEC scheme on the chosen microcontroller (Atmel AVR Xmega128 A3 [11]) and measured the clock cycles of executing encoding and decoding. Using our simulation data, we could determine the proportion of each code and scaled the energy consumption according to the microcontrollers clock cycles. Table II shows the important parameters of the FEC codes, which are used in the following calculations, where N refers to the block length of the code K denotes the message length, t signifies the correctable symbols and κ_4 refers to the energy consumption of coding 1 bit using the particular FEC.

TABLE II: Summary of FEC code parameters

Code	Complexity	Type	N	K	t	κ_4
No FEC	none	none	1	1	0	0
Hamming (255,247)	low	block	255	247	1	5.0522E-9
Reed-Solomon (511,501)	high	block	511	501	5	5.4344E-7
BCH (511,502)	high	block	511	502	4	1.7619E-5

IV. Optimal wakeup signal length

This section shows how to determine the optimal wakeup signal length to minimize the energy needed by one node, in order to maximize its lifetime. In case no aggregation is used, the nodes send as many separate packets (consisting of ω header and φ payload), as necessary to send the information. On the contrary, in case of aggregation the length of the packet depends on the number of φ payloads, the used n aggregation number and the $FEC(N, K)$ code applied. Formula (1) determines the packet length:

$$\sigma = \begin{cases} \omega + N \left\lceil \frac{n\varphi}{K} \right\rceil & \text{, if aggregation is ON} \\ \omega + \varphi & \text{, if aggregation is OFF} \end{cases} \tag{1}$$

According to the previous considerations, in case of aggregation only one packet is sent, otherwise as many as the n aggregation number. Therefore λ_t also depends on aggregation according to Formula (2):

$$\lambda_t = \begin{cases} 1 & \text{, if aggregation is ON} \\ n & \text{, if aggregation is OFF} \end{cases} \tag{2}$$

Equation (2) does not contain the case, when a node sends more aggregated packets in the same T period. The amount of time necessary for sending one normal packet is:

$$t_{packet} = \frac{\sigma}{B}. \tag{3}$$

The amount of time necessary for sending an ACK packet can be expressed as:

$$t_{ACK} = \frac{\omega_h}{B} = \frac{128 \text{ bit}}{9600 \text{ bit/s}} = 13.33 \text{ ms}. \tag{4}$$

The quality of the radio channel is modeled by the Bit Error Rate (BER), which gives the number of damaged bits per all sent bits ratio. In this paper we do not consider bit deletion errors in the channel.

In the calculation of PER we assume, that some kind of FEC is applied to correct statistically independent bits of the corrupted packet, and some kind of Message Authentication

Code (MAC) is used to recognize malicious modifications of the payload. This paper does not take correlated bit errors into account. We also assume that FEC is not applied to the header of the packets so that no unnecessary calculations are made in case the destination address was corrupted. In order to calculate the amount of Packet Error Rate (PER) of the channel in case of using FEC codes, we have to take into account the t error correction capabilities of the FEC codes, where β defines the amount of corrected bits in the payload:

$$\beta = \sum_{i=0}^{t} \binom{N}{i} BER^{i} \left(1 - BER\right)^{N-i}$$

$$PER = 1 - \left(\left(1 - BER\right)^{\omega_h} \beta^{\left\lceil \frac{n\varphi + \omega_{MAC}}{K} \right\rceil} \right) \tag{5}$$

Without the use of FEC (5) is simplified to (6), as the values of the parameters are $N = 1$, $K = 1$ and $t = 0$ according to Table II.

$$PER_{NoFEC} = 1 - (1 - BER)^{\omega_h + n\varphi + \omega_{MAC}} \tag{6}$$

The packets transmitted are received successfully with probability $1 - PER$ on a channel characterized by a certain PER. The probability, that the number of retransmissions until success will be k, is given by probability variable X with geometric distribution and $p = 1 - PER$

$$P(X = k) = PER^{k-1}(1 - PER) \tag{7}$$

The expected value of X – which denotes that how many packets need to be sent for a successful reception in an average – can be expressed as (by geometric distribution):

$$E(X) = \sum_{k=1}^{\infty} k \; P(X = k) = \frac{1}{1 - PER} \tag{8}$$

Using (8), the average number of required retransmissions can be determined. The value of r should be a positive integer ($r \in \mathbb{Z}^{+}$), because every fraction of packet sent is considered to be a part of a new packet, therefore:

$$r = \left\lceil \frac{1}{1 - PER} \right\rceil \tag{9}$$

The total amount of time a node spends in transmission state is:

$$t_{tx} = \lambda_t(\Delta_{wake} + r \; t_{packet}) + r \; \lambda_r \; t_{ACK}. \tag{10}$$

To describe the total amount of time spent with receiving a packet a probability variable Y is introduced, which describes how much time is left from time interval Δ_{wake} at the moment the receiver node listens to the radio channel. Y is assumed to be uniformly distributed, because the amount of time left from Δ_{wake} can have any value from 0 to Δ_{wake} with equal probability. Therefore the expected value of Y probability variable is $E(Y) = \frac{\Delta_{wake}}{2}$. This is why the receiver node has to spend $E(Y)$ time (after a successful wake) in receiving state, while the packet sending starts. According to the previous considerations the amount of time spent with receiving is:

$$t_{rx} = \lambda_r \left(\frac{\Delta_{wake}}{2} + r t_{packet} \right) + r \lambda_t (t_{WaitForACK} + t_{ACK}). \tag{11}$$

The time spent for listening to the wakeup signal in interval T becomes

$$t_{\sum listen} = \left\lfloor \frac{T - t_{tx} - t_{rx}}{\mu + t_{listen}} \right\rfloor t_{listen}. \tag{12}$$

To catch the wakeup signal every time the listening period is:

$$\mu = \Delta_{wake} - 2 \; t_{listen}. \tag{13}$$

The time spent in sleep state from period T is, what remains:

$$t_{sleep} = T - t_{rx} - t_{tx} - t_{\sum listen}. \tag{14}$$

The amount of electric charge used in a second can be calculated as:

$$C_{req} = \frac{t_{tx} i_{tx} + (t_{rx} + t_{\sum listen}) i_{rx} + t_{sleep} i_{idle}}{T}. \tag{15}$$

To compare the benefits due to the use of aggregation, FEC and the optimal wakeup signal length combined we propose to use the lifetime of the nodes expressed in days, calculated as:

$$\eta = \frac{C_{src}}{T \; C_{req}}. \tag{16}$$

To maximize the lifetime, it is equivalent to minimize the energy used in a second:

$$\max_{\Delta_{wake}} \{\eta\} \equiv \min_{\Delta_{wake}} \{C_{req}\}. \tag{17}$$

V. RESULTS

This section shows the results based on the calculations derived in the previous section. To be able to show some connections between the parameters, a sample scenario was chosen using the hardware and protocol introduced in the previous sections. The value of number of receptions λ_r was set to 1, and the channel was considered to be average quality with $BER = 4 \cdot 10^{-4}$.

Figure 2 depicts the change in expected lifetime (in days) as a function of Δ_{wake} (in seconds) with a fixed aggregation number of $n = 10$. Table III shows the optimal Δ_{wake} and maximal η for Figure 2. It can be observed, that the graphs from different scenarios have a maximum, therefore in every case the optimal Δ_{wake}, which maximizes the lifetime can be determined for every BER and aggregation number n. The value of the optimal Δ_{wake} is to the third decimal the same in case of different FEC codes, provided that the BER of the channel stays the same. In case of a lower quality channel (higher BER) the benefits of using the optimal wakeup signal length are more significant while enabling FEC as well. The benefit of using the optimal Δ_{wake} could result at least twofold increase in lifetime in case of no forward error correction used and over a fivefold increase in lifetime using RS or BCH FEC codes with the previously mentioned n and BER values.

Remark. In case of a poor channel, the value of Δ_{wake} depends more on the aggregation number, as the node is forced to resend the packets more times.

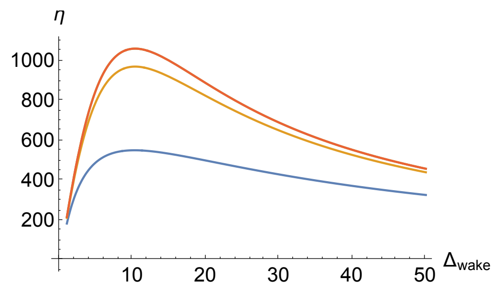

Fig. 2: The optimal value of Δ_{wake} is the same in case of different FEC codes with $BER = 4 \cdot 10^{-3}$

TABLE III: Optimal Δ_{wake} and maximal η for Figure 2

FEC	n	Δ_{wake}	η
No FEC	10	10.246	551.726
Hamming	10	10.312	971.346
BCH	10	10.319	1061.42
RS	10	10.319	1061.42

The optimal Δ_{wake} as a function of n in case of different BER channels can be seen in Figure 3. Δ_{wake} is not linearly depending on n, but the poorer the BER of the channel, the optimal Δ_{wake} decreases more rapidly as n grows.

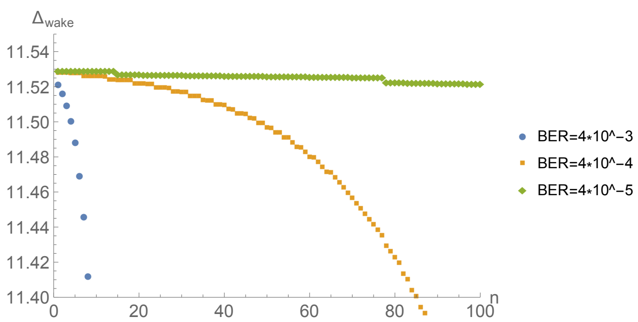

Fig. 3: Δ_{wake} as a function of n in case of different BER channels

Next we show that applying FEC code to the data increases the importance of choosing the optimal wakeup length beyond the significant energy benefits. Figure 4 shows that in case of no FEC is used, as the amount of aggregation is raised, the effect of the wakeup signal length increases as well. The lifetime scales very poorly in case the payload is increasing. Table IV shows the optimal Δ_{wake} and maximal η for Figure 4.

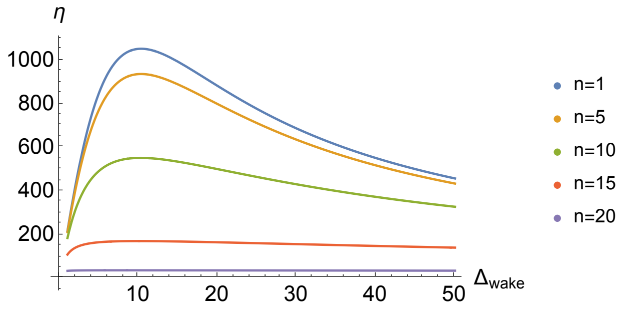

Fig. 4: η as a function of Δ_{wake} in case of different amounts of aggregation without FEC codes

TABLE IV: Optimal Δ_{wake} and maximal η for Figure 4

n	Δ_{wake}	η
1	10.318	1052.39
5	10.308	936.465
10	10.246	551.726
15	9.907	170.23
20	7.883	36.1383

Figure 5 and its corresponding Table V however, which shows the same aggregation values shows that as the benefits of using FEC as aggregation increases, the length of the wakeup signal determines the efficiency of the solution. In this case the lifetime scales significantly better as the amount of aggregated data is increasing.

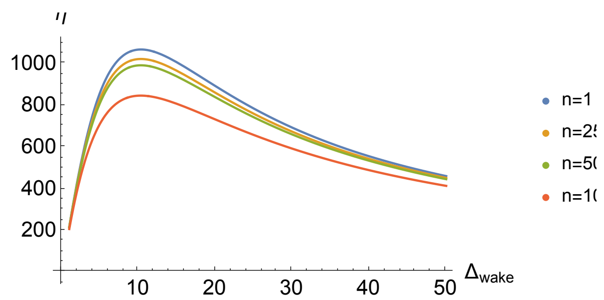

Fig. 5: η as a function of Δ_{wake} in case of different amounts of aggregation with BCH FEC code

TABLE V: Optimal Δ_{wake} and maximal η for Figure 5

n	Δ_{wake}	η
1	10.32	1067.07
25	10.317	1021.52
50	10.315	991.262
100	10.303	846.002

The same effect can be observed in case of Reed-Solomon and Hamming codes on Figure 6 and Figure 7 and their corresponding Table VI and Table VII.

Remark. Despite the magnitude of the effect on efficiency, the optimal wakeup signal length stayed the same in case of different amounts of aggregation or various FEC codes.

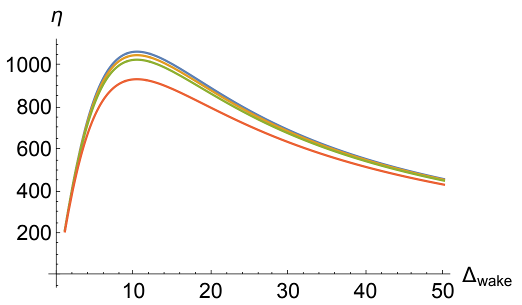

Fig. 6: η as a function of Δ_{wake} in case of different amounts of aggregation with Reed-Solomon FEC code

TABLE VI: Optimal Δ_{wake} and maximal η for Figure 6

n	Δ_{wake}	η
1	10.32	1067.07
25	10.319	1050.32
50	10.317	1028.79
100	10.311	935.817

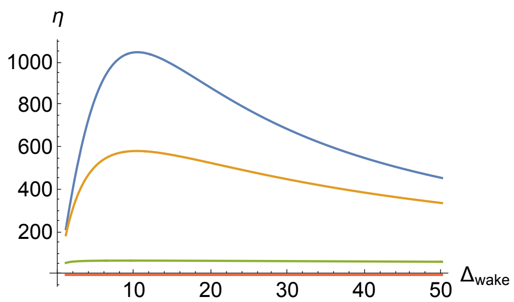

Fig. 7: η as a function of Δ_{wake} in case of different amounts of aggregation with Hamming FEC code

TABLE VII: Optimal Δ_{wake} and maximal η for Figure 7

n	Δ_{wake}	η
1	10.318	1049.57
25	10.259	584.053
50	9.245	67.553
100	1.001	0.313514

Let us examine how different FEC codes influence the lifetime without the use of aggregation. The results are shown in Figure 8, which makes it clear, that not using FEC codes gives the shortest lifetime. Using any kind of FEC improves lifetime, but Reed-Solomon and BCH codes perform better. In this case n refers to the number of packets sent, because no aggregation was used. It can be seen, that in case of higher packet number, the result with the FEC codes do not differ much from the case with no FEC, since all of them converge to 0. This is because the packet number is so high, that due to repetitions necessary on the poor channel, the nodes are almost permanently awake. However, different FEC codes reach this state at distinct n values.

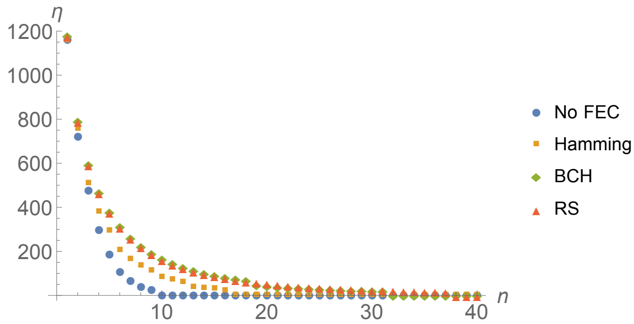

Fig. 8: Comparison of FEC codes without aggregation

Figure 9 investigates the benefits of using different FEC codes in case if aggregation enabled. The curves are decreasing, because longer packets need more energy. The worst results are in case we did not use FEC, and the best FEC code was Reed-Solomon. BCH is close to RS, but RS can correct one more bit, which explains its better values. The previous two graph shows clearly that the use of aggregation is very efficient, as it increases lifetime significantly. As the aggregation number n increases, the use of FEC codes is also more efficient.

Remark. We have investigated to include an RTS-CTS mechanism to our calculations, but the final results were not significantly affected by the use of RTS-CTS, therefore we omitted them to simplify the formulas.

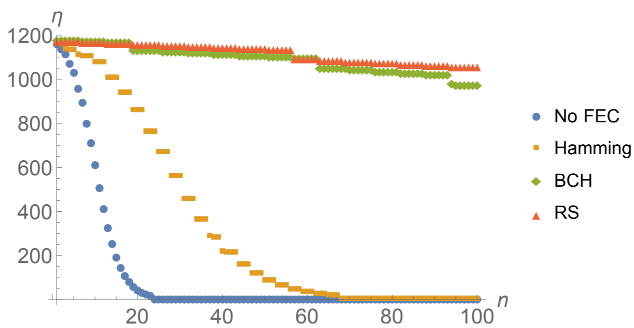

Fig. 9: Comparison of FEC codes with aggregation enabled

Combining all previous diagrams and the investigated phenomena Figure 10 shows the connection between η expected lifetime, n aggregation number, and Δ_{wake} optimal wakeup signal length in case of a poor $BER = 4 \cdot 10^{-3}$ channel. We can observe that the expected lifetime of a system without FEC stays much lower, than using FEC codes. The optimal length of the wakeup signal is depending on the type of FEC in use and the amount of aggregation.

Remark. The 3D plot should be continuous, as variable Δ_{wake} is not an integer, however it is easier to understand it on a discrete plot.

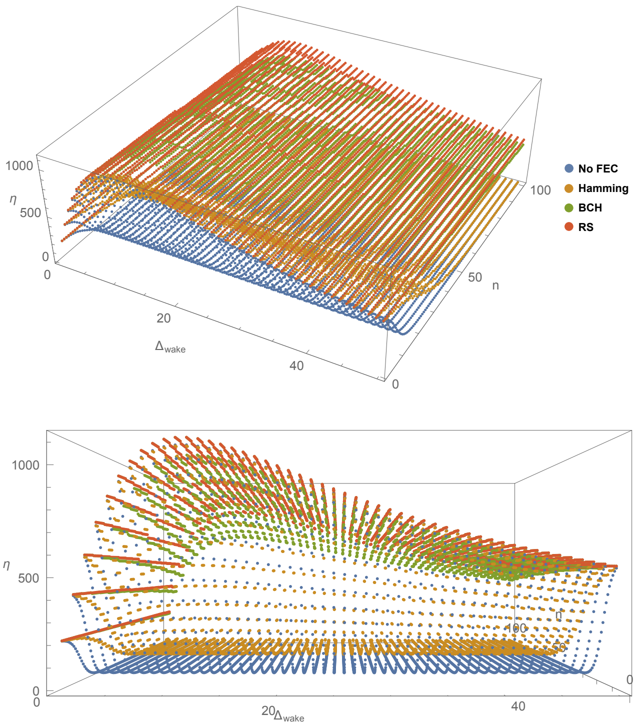

Fig. 10: The expected lifetime as a function of n and Δ_{wake}

VI. Conclusion

In this paper the authors presented the sleep-wake cycle optimization problem in WSNs. The authors showed, that (see Figure 3), the optimal value of Δ_{wake} wakeup signal length for minimal power consumption is not significantly affected by the length of the payload in case of a good quality channel. Besides the value of the optimal Δ_{wake} wakeup signal length is very similar in case of the presented block forward error correction codes, therefore it is considered independent. The results showed, that FEC codes and aggregation should be both combined with the optimal Δ_{wake} to significantly increase the lifetime of the devices. Solving both optimization problems gives the best results: first determine the optimal value of Δ_{wake} for a certain BER, because it is independent of the other parameters; then calculate the optimal payload aggregation number according to the channel bit error rate (shown in [18]). Utilizing only the optimal wakeup signal length we can extend the battery life of a node more than twofold (depending on the BER of the channel and the amount of aggregation) without using FEC codes (as presented on Figure 4). In case the radio channel has poor quality, FEC codes combined with the optimal wakeup signal length can prolong lifetime of the nodes up to four times compared to baseline (as presented on Figure 5). Further investigations could dive into the problem of packet loss and collision with other packets on the radio channel to extend the presented results.

Acknowledgment

This research has been supported by BME-Infokom Innovator Nonprofit Ltd., http://www.bme-infokom.hu.

References

[1] L. Hanzo, H. Haas, S. Imre, D. O'Brien, M. Rupp, and L. Gyongyosi, "Wireless Myths, Realities, and Futures: From 3G/4G to Optical and Quantum Wireless," *Proceedings of the IEEE*, vol. 100, no. Special Centennial Issue, pp. 1853–1888, May 2012.

[2] A. Milankovich, G. Ill, K. Lendvai, S. Imre, and S. Szabo, "Wakeup signal length optimization combined with payload aggregation and FEC in WSNs," in *European Wireless 2015; 21th European Wireless Conference; Proceedings of*, May 2015, pp. 1–6.

[3] G. A. Elkheir, A. S. Lioumpas, and A. Alexiou, "Energy efficient cooperative scheduling based on sleep-wake mechanisms," in *Wireless Communications and Networking Conference Workshops (WCNCW), 2012 IEEE.* IEEE, 2012, pp. 238–242.

[4] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE.* IEEE, 2002, pp. 1567–1576.

[5] C.-C. Hsu, M.-S. Kuo, S.-C. Wang, and C.-F. Chou, "Joint Design of Asynchronous Sleep-Wake Scheduling and Opportunistic Routing in Wireless Sensor Networks," *Computers, IEEE Transactions on*, vol. 63, no. 7, pp. 1840–1846, 2014.

[6] M. Magno and L. Benini, "An ultra low power high sensitivity wake-up radio receiver with addressing capability," pp. 92–99, 2014.

[7] J. Ansari, D. Pankin, and P. Mähönen, *Radio-Triggered Wake-ups with Addressing Capabilities for extremely low power sensor network applications.* IEEE, 2008.

[8] H. Karvonen, J. Petajajarvi, J. Iinatti, M. Hamalainen, and C. Pomalaza-Raez, "A generic wake-up radio based MAC protocol for energy efficient short range communication," in *Personal, Indoor, and Mobile Radio Communication (PIMRC), 2014 IEEE 25th Annual International Symposium on.* IEEE, 2014, pp. 2173–2177.

[9] P. Kamalinejad, K. Keikhosravy, M. Magno, S. Mirabbasi, V. C. M. Leung, and L. Benini, "A high-sensitivity fully passive wake-up radio front-end for wireless sensor nodes," in *Consumer Electronics (ICCE), 2014 IEEE International Conference on.* IEEE, 2014, pp. 209–210.

[10] Texas Instruments, Incorporated, *CC1101 Low-Power Sub-1 GHz RF Transceiver (Rev. I)*, Nov. 2014.

[11] Atmel, "8/16-bit XMEGA A3 Microcontroller," pp. 1–134, Jun. 2013.

[12] S. Lin and D. J. Costello, *Error Control Coding*, ser. Fundamentals and Applications. Prentice Hall, 2004.

[13] T. Etzion and A. Vardy, "Perfect binary codes: constructions, properties, and enumeration," *IEEE Transactions on Information Theory*, vol. 40, no. 3, pp. 754–763, May 1994.

[14] M. P. C. Fossorier, S. Lin, and J. Snyders, "Reliability-based syndrome decoding of linear block codes," *IEEE Transactions on Information Theory*, vol. 44, no. 1, pp. 388–398, Jan. 1998.

[15] D. V. Sarwate and N. R. Shanbhag, "High-speed architectures for Reed-Solomon decoders," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 9, no. 5, pp. 641–655, Oct. 2001.

[16] S. B. Wicker, V. K. Bhargava, I. C. Society, and I. I. T. Society, *Reed-Solomon codes and their applications*. Inst of Electrical &, Jun. 1994.

[17] R. C. Bose and D. K. Ray-Chaudhuri, "Further results on error correcting binary group codes," *Information and Control*, vol. 3, no. 3, pp. 279–290, Sep. 1960.

[18] Lendvai, K, Milánkovich, Á, Imre, S, and Szabo, S, "Optimized packet size for energy efficient delay-tolerant sensor networks with FEC," in *Proceedings of the 12th International Conference on Telecommunications (CONTEL '13)*.

Gergely Ill received his M.Sc. degree in technical informatics from the Budapest University of Technology and Economics in 2012. He continued his Ph.D. studies at the Department of Networked Systems and Services. His research interests includes the energy efficiency of wireless sensor networks, wireless protocols and their security aspects.

Ákos Milánkovich received his M.Sc. degree in technical informatics in the field of networks and services from the Budapest University of Technology and Economics in 2012. He continued his Ph.D. studies at the Department of Networked Systems and Services. His research interests include wireless sensor networks, positioning technologies and UWB.

Károly Lendvai received his degree in electrical engineering in 2008 at BUTE. At the Corvinus University of Budapest he graduated as engineer economist in 2011. He started his Ph.D. studies in 2008 at the Department of Telecommunications. His main research areas are optimization of transport protocols in wireless environment, wireless sensor networks and modeling of user movement in mobile networks.

Sándor Imre was born in Budapest in 1969. He received the M.Sc. degree in Electronic Engineering from the Budapest University of Technology (BME) in 1993. Next he started his Ph.D. studies at BME and obtained dr. univ. degree in 1996, Ph.D. degree in 1999 and D.Sc. degree in 2007. Currently he is carrying his teaching activities as Head of the Dept. of Networked Systems and Services of BME. His research interest includes mobile and wireless systems, quantum computing and communications. Especially he has contributions on different wireless access technologies, mobility protocols and reconfigurable systems.

Sándor Szabó was born in 1977. He graduated in 2000 at the Faculty of Electrical Engineering and Informatics and received his Ph.D. degree in 2011. Currently he works as adjunct at the Dept. of Networked Systems and Services. He takes part in research projects at the university and he is the head of the Mobile Innovation Center. His research areas are integration of wired and wireless networks, mobility handling methods, IMS systems and sensor networks.