International Journal of Advances in Computer Networks and Its Security
Volume 6 : Issue 1 [ISSN 2250-3757]
Publication Date : 18 April, 2016

Detailed estimation of battery life in a WSN deployment in the field

David Chaves-Dieguez, Felipe Gil-Castiñeira

***Abstract* — This document presents a detailed analytical and practical method for the estimation of battery life expectancy of a multi-hop mesh network. This network is composed by a set of MicaZ nodes using the XMesh routing protocol. The energy drawn for each of the tasks performed by the nodes is measured, and network topology and radio propagation are characterized in order to obtain accurate predictions of the lower bound for the battery duration of the nodes. This estimation is of paramount importance for field deployments, where energy is a scarce resource and replacing node batteries is a slow and costly process that can disable the network for relatively long periods of time.**

***Keywords*—WSN, battery, power, MicaZ, XMesh**

I. Introduction

Wireless sensor networking (WSNs) is a key enabling technology that has led to important technological advances in several fields, including environmental and structural monitoring [1][2], healthcare [3], science and engineering [4][5], etc.

The development of devices of reduced dimensions, with reduced battery requirements and that are capable of autonomously performing computations and communicating via radio interfaces has provided the needed tools to create systems able to monitor, control and interact with their environment. With the help of these systems, the implementation of smart environments, environments that are able to react to external stimuli and perform advanced tasks, have been made possible.

However, although the previously mentioned devices have reduced requirements, there are still challenges regarding the limited resources that are available in certain practical applications.

Specifically, battery is one of the more critical resources in environmental monitoring applications. These applications are characterized by low requirements in terms of computational power, network performance and memory resources. However, the availability of continuous and reliable power sources for long time periods in the field is usually scarce. This makes energy optimization and battery life prediction a critical task in the planning of a field deployment.

David Chaves-Dieguez
Universidade de Vigo
Spain

Felipe Gil-Castiñeira
Universidade de Vigo
Spain

This paper presents a detailed energy consumption analysis of a WSN deployed in the field and composed of MicaZ nodes, whose objective is to perform environmental monitoring on a defined area. Multiple aspects regarding the operation of the nodes have been taken into account in order to provide a characterization of the sources of energy consumption as detailed as possible, including current drawn from sensors, radio communications, processor and, especially, the energy dedicated to routing and forwarding tasks, which is the main source of energy usage in large mesh networks with low sensing frequencies.

The rest of the paper is structured as follows: section II presents some of the works in the state of the art that are related to the results presented; section III briefly describes the hardware configuration of the analyzed nodes and their main programming; section IV presents the detailed current draw measurements that have been made in order to characterize each task performed by the nodes; section V takes the results of the previous sections and present the calculations made in order to predict battery life together with the main obtained results; and finally section VI concludes the paper and presents some recommendations to optimize energy resources in large deployments that can be derived from the presented results.

II. Related Work

MicaZ nodes have been used both for laboratory testing and for long-term deployments, as demonstrated by Ali et al. [6], and shown in testbeds described by several related publications such as those by Navarro et al. [7] or Paek et al. [8]. However, for this kind of deployments to be usable in practical applications, an accurate estimation of battery life and energy consumption is needed.

The energy performance of AODV, DYMO and XMESH routing protocols for 802.15.4 wireless sensor networks as a function of duty cycle has been analyzed by Kumar and Tiwari [9], although their results are only based on extensive simulation of the nodes and do not take into account any specific topology distribution.

Davis et al. [10] have examined energy consumption characteristics and communications performance of a MicaZ XMesh network deployment both in laboratory and field settings, and it states that motes closer to the base station present a reduced battery life compared to those at the edge of the network, due to the larger amount of packets that they need to forward. Also, the topic of battery life expectancy is also briefly covered by Teo et al. [11], but only taking into account a simplified duty cycle approach. However, the detailed analysis of the power consumption of each task performed by a node in the network as the one presented in this paper is outside the scope of their work.

Regarding radio propagation characteristics of the MicaZ nodes, the results provided by Su and Alzaghal [12] offer some conclusions about the radio performance of the devices that will be taken into account in the corresponding sections of this article in order to estimate maximum reliable communication distances for the deployment, together with the field measurements that were performed. Although radio propagation does not directly affect energy consumption, it does affect network node density, and as a consequence the maximum number of messages that will have to be forwarded, hence its importance in the results of this work.

III. WSN nodes

A. Hardware setup

In order to monitor the environmental variables on the targeted area of deployment, the MicaZ nodes from Crossbow (currently from Memsic [13]) have been selected due to their availability and extensive usage in academic research. Also, the fact that they are supported by the open source operating system TinyOS [14] is an advantage in terms of implementation flexibility and community support.

Nodes use a CC2420 chipset, which is based on an ATMEL ATMega128L processor, with 128KB of program Flash memory, 512KB of sample Flash memory, and 4KB of EEPROM for configuration data. Their radio connectivity is provided by a low power IEEE802.15.4 interface.

A MPR2400CA [13] sensor board is attached to each node. This board allows nodes to sample light intensity, temperature, ambient humidity, barometric pressure and 2-axis acceleration. However, due to the fact that the objective of the deployment is to monitor environmental variables, this last sensor will not be used.

In order to be able to program sensor nodes and act as a data interface with the infrastructure, a MIB520 board will be attached to a PC, thus acting as a base station and gateway in the deployment.

B. Sensing application and XMesh stack

In order to perform the proposed ambient monitoring tasks, an application that runs on the sensor nodes was developed. Since the main objective of such application is to gather the information and send it to a central server, and the data is not critical (i.e. packet losses are easily tolerated), the design of the application was as simple as possible in order to optimize energy resources.

The gateway node will be powered through its connection with a PC, so its energy resources are not limited in the sense of the analysis performed in this work. Therefore its operation will not be detailed here.

The application in each node consists on a loop that is repeated in constant and defined intervals. Each iteration of that loop, as shown in Fig. 1, performs a sequence of measurements covering all the used sensors (humidity, pressure, temperature and light) plus the battery level of the device, and then sends one message with all the measurements through the radio interface.

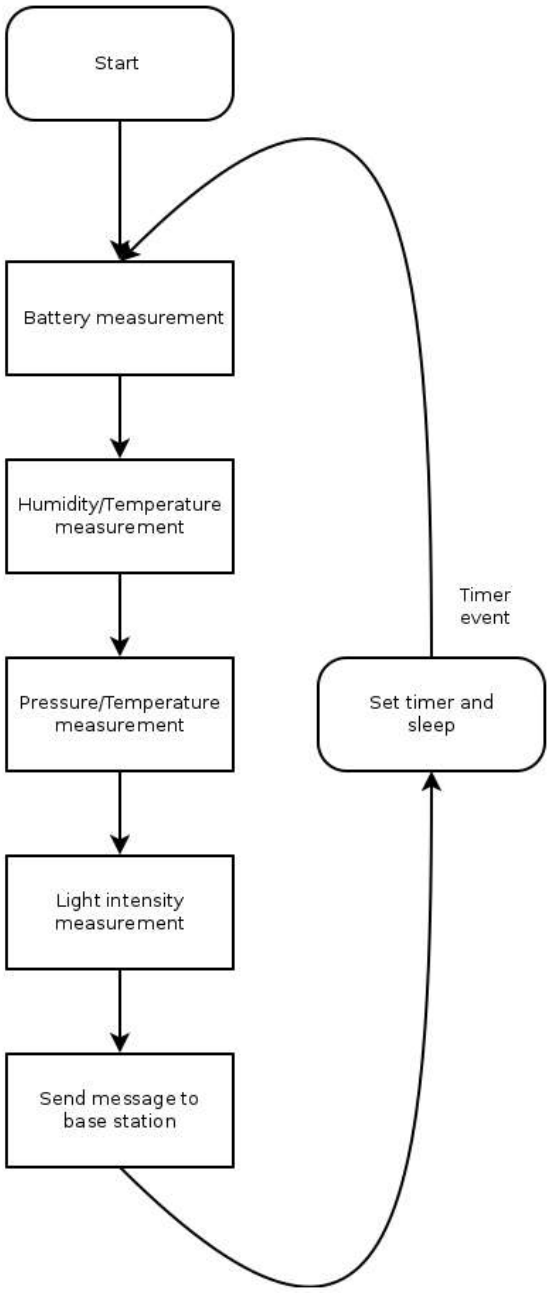

Figure 1. Main loop of the application

In order to be able to send the messages to the base station, the XMesh protocol is used to provide an ad-hoc multi-hop mesh network with a reduced energy consumption. This network configuration provides the self-healing property to the deployment, allowing for a number of nodes to malfunction and still be able to route information to the base station. In this configuration, every node in the network will act at the same time as a data producer and as a router, forwarding packets produced by neighbor nodes towards their recipient. Although this increases the fault tolerance of the network as a whole, it also increases energy requirements since there are more messages that have to be transmitted and radio interfaces have to be more time in receive mode.

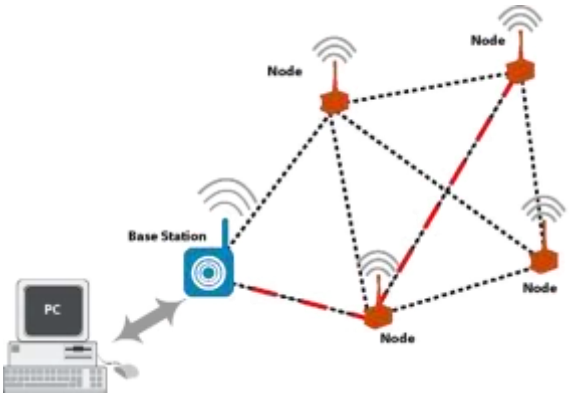

Figure 2. XMesh network in mesh mode

As opposed to many other mesh routing protocols, the XMesh protocol improves battery life of routing nodes by not requiring their radio interface to be continuously active in order to receive messages coming from adjacent neighbors. Instead, routing nodes can periodically deactivate their radio without affecting network error rates by applying the following procedure:

- Any node trying to send a message to a neighbor must first send a radio preamble of duration T_0, and the message payload immediately after.
- Routing nodes activate their radio interface in receive mode with a period shorter than T_0, and for a very short time that must be enough to detect the presence or a radio preamble.
- If a preamble is not detected, they deactivate their radio modules for the next period.
- If a preamble is detected, they wait for it to end and then receive the message payload.

The key for this procedure to work is the radio monitoring period T_0. Since the duration of the preamble is longer than the time between consecutive activations of the receiving radio interface in routing nodes, it is guaranteed that it will be detected by nodes in the radio coverage of the transmitter, hence the subsequent message will be received.

In addition to the data messages, there are two other types of messages in the XMesh protocol that will be taken into account for this description: route updates and health updates. Route updates are part of the routing protocol in XMesh and allow to determine the cost of each link between nodes in the network. These messages are sent in broadcast mode and are not routed. The period of their emission is a parameter that has to be chosen carefully, since a value too low can negatively affect network performance (flooding), but a value too high will increase the time needed for the network to recover from node failures or to achieve a stable working topology. Health updates are sent to the base node and provide information about the operating state of each node in the network (battery level, link quality with respect to their neighbors, message transmission statistics, etc.).

Since health updates are not needed for the intended application and they suppose an additional source of energy consumption, they will be disabled for the performed tests. Also, taking into account the nature of the measured variables, the loss of a reduced number of messages is not a problem for the intended application, so end to end acknowledgements will also be disabled in order to increase battery life.

Integrity checks are performed for each received message in every node in the network, and if an error is detected the payload is transmitted again up to 8 times in order to provide hop by hop reliability.

IV. Current draw measurements

After having described with a high level of abstraction the operation of the monitoring application and the network protocols involved in the deployment, a more detailed analysis of the current drawn by each of the subtasks comprised in an operating cycle of each node. Both the application and the network layer are supported by the TinyOS embedded operating system, and some of the properties of this system will have an impact in some of the used considerations.

In order to determine the current used by each of the nodes in the network for each performed task and later be able to estimate battery life, a resistor with a fixed low value was connected in series with the main power of a test node, and the voltage drop in that resistor was measured with a high precision digital oscilloscope, thus allowing to measure both current draw and operation durations.

A. *Base timer and sleep mode*

MicaZ nodes can be put in a deep sleep mode in which most of their functions are disabled, including radio interfaces, and only external interruptions and main timer events are detected by the processor. In TinyOS this mode is directly controlled by the operating system and cannot be forced by applications. It drives every software timer and task scheduling, its operating period is 225ms and its current signature is show in Fig. 3.

Each timer activation consists on a 7.14mA current draw period lasting 360µs, together with a 0.67mA current draw period lasting 2.4ms and a 1.34mA current draw period lasting 2.2ms. However, since this timer is never stopped or delayed as long as the node is active, a measured constant value of 42µA will be used as the baseline current draw for the powered node in the calculations.

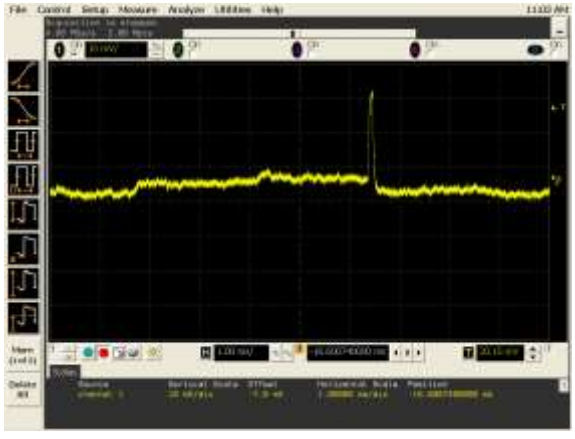

Figure 3. Base timer current draw

B. Sensor board

Powering the MTS400CA sensor board also has an effect on the overall current draw of the node, despite the sensors not being used. This current draw takes the form of exponentially decreasing peaks with a period of 48ms and whose signature can be seen on Fig. 4. Taking into account the falling curve, its maximum value and its duration, the current draw of this board is equivalent to a 8mA draw lasting 300µs once every 48ms.

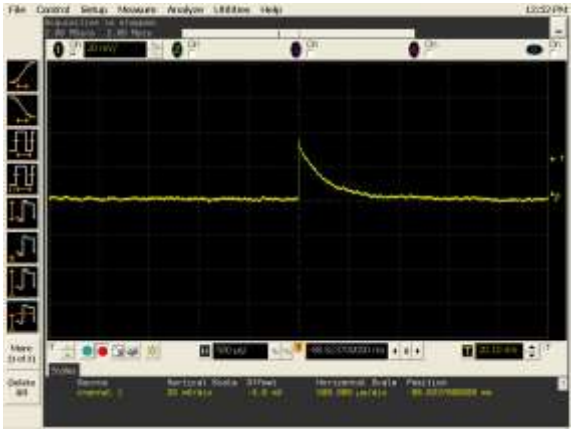

Figure 4. MTS400CA sensor board current draw

C. Sensor sampling

For each of the samples taken in each iteration of the main application loop its current draw will be detailed in this section. Since base timer and preamble sampling activities cannot be disabled during the measurements, for some of the included graphics, periodic peaks in the signals shown will be ignored in this section as they are related either to the base timer described in section IV.A or to the radio preamble sampling activities that will be described in section IV.D.

Fig. 5 shows the battery sampling step, which starts with a 4.7ms period in which the processor is active, drawing about 8.8mA in average. Then, current draw falls to 1.1mA for about 325ms, until the sample from the sensor is obtained and processed.

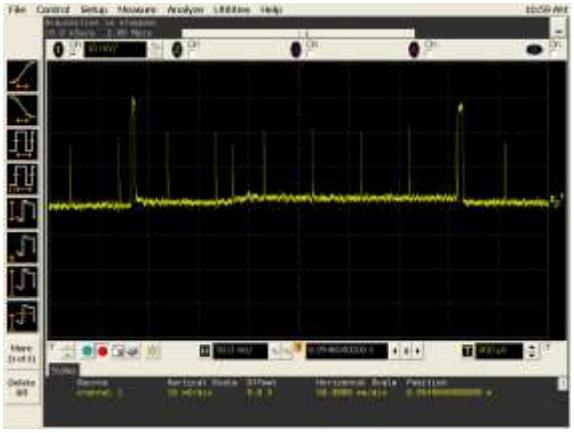

Figure 5. Battery sampling current draw

The Sensirion SHT11 temperature and humidity sensor produces a current draw while sampling as shown in Fig. 6. This current drawing period starts with a microprocessor activity period lasting 8.5ms and drawing 8.8mA, followed by a 100ms interval with a current draw of 0.5mA and a final 250ms period with a 1mA draw.

Figure 6. Sensirion SHT11 temperature and humidity sensor current draw

The Intersema MS55ER barometric pressure and temperature sensor has a current draw signature composed of several peaks while active, as shown in Figure 7. The measurement period starts with a microprocessor activity period of 13.5ms and 8.8mA, followed by a 350ms period containing one current peak every 23ms.

Figure 7. Intersema MS55ER pressure and temperature sensor current draw

Each one of those peaks is as shown in Figure 8, whose current signature can be conservatively approximated by a 2.5ms period of 1.25mA draw period lasting 2.5ms and a 7mA draw period lasting 1.75ms.

The TAOS TLS2550 sensor samples two photodiodes and returns a value of the light intensity perceived. Its current draw is a constant of 0.6mA over its activity period of 835ms per measurement. That period starts with a processor activity of 8.8mA for 10.5ms and ends with another 8.8mA for 4.5ms.

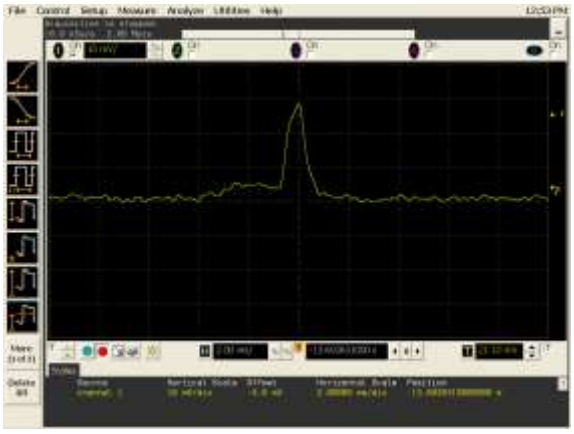

Figure 8. Intersema MS55ER pressure and temperature sensor current draw (peak detail)

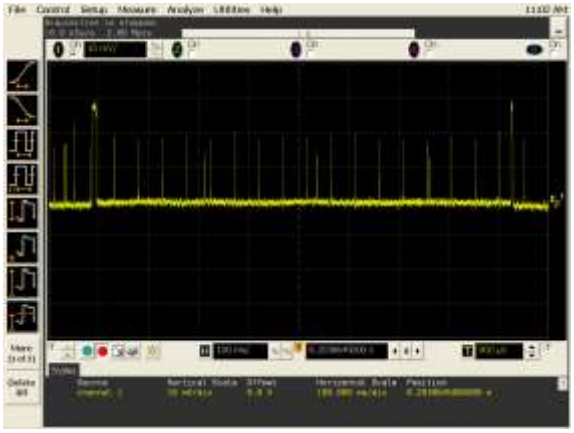

Figure 9. TAOS TLS2550 sensor current draw

D. *Radio interface*

The radio interface of the nodes performs three main activities, namely: message transmission, message reception and the periodic radio preamble sampling previously described. Although message transmission and reception depend basically on the defined sampling intervals and the network topology (nodes act as routers and hence must forward messages from neighbors), radio preamble sampling is a task with a fixed period and can be one of the main sources of energy consumption in low transmission rate networks.

The radio preamble sampling takes place once every 122ms, and has a current signature as shown in Fig. 10.

Since this is the current draw signature that is most frequently repeated in the analyzed scenario, it will be characterized in detail by dividing it in three sections. The first one will be approximated by a constant current draw of 7mA lasting 1.8ms. The second one will be conservatively approximated by a constant current draw of 25mA lasting 1ms. Finally, the last section will be approximated by a decreasing exponential with a time constant of 440µs, which taking into account its maximum value and falling time is equivalent to a constant current draw of 29mA lasting 440µs.

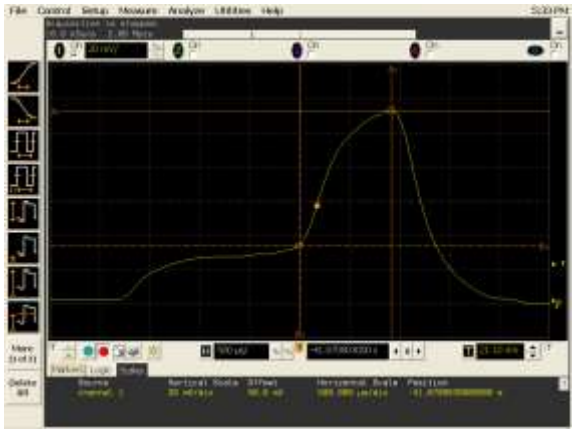

Figure 10. Radio preamble sampling

Each transmitted message draws a current shown in Fig. 11. Although the transmission of the payload takes less than the time shown in that graph, the sending of the preamble, the random back-off period and the link level acknowledgement increase the radio activity period. On average, each transmitted message draws 23mA for 145ms when transmitting at maximum output power (0dBm for the CC2420).

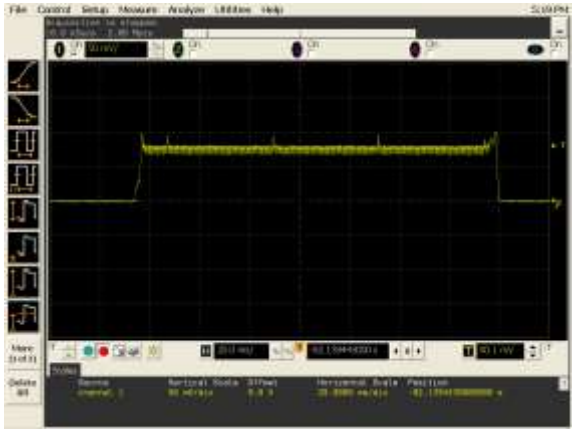

Figure 11. Message transmission at 0dBm

If the transmission power is lowered to the minimum that the CC2420 can handle (-25dBm), the current draw descends only to 20mA, while the radio range decreases considerably. Due to this fact, the deployment will use the maximum transmission power in order to reduce both the number of nodes needed to cover the intended area and the hence the number of forwarded messages.

Radio message reception has a variable duration due to the lack of synchronicity between message transmission and preamble sampling previously explained. However, that duration has an upper limit of 145ms, which is the duration of the whole message transmission as shown before. For this maximum duration, the current drawn is as shown in Fig.12 that will be modelled as a constant value of 19mA on average.

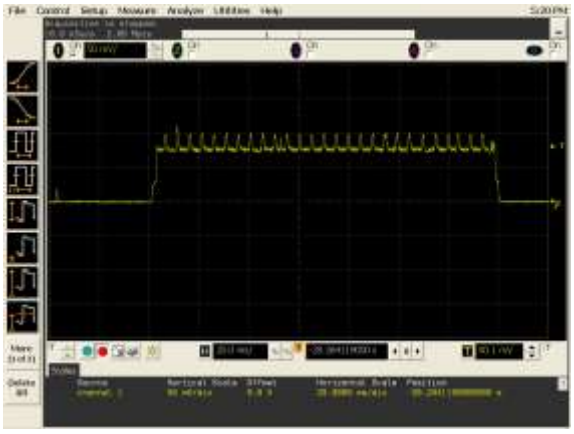

Figure 12. Message reception

v. Constraints and calculations

A. Environment characterization

In order to perform tests in a real deployment in the field, an area of approximately 2500m^2 in Vigo (Spain) was selected. This area is shown in Fig. 13 and it is surrounded by forest tracks.

Figure 13. Test deployment area

Several radio propagation measurements have been performed in the designated area. According to these measurements, the signal power received by a node as a function of the distance between it and a 0dBm transmitter follows in average the function described in (1), where x denotes the distance between emitter and receiver and P is measured in dBm.

$$P(x) = -40.2251 - 2.341 * 10 * \log_{10}(x) \quad (1)$$

The background noise in this area has also been measured, including the interference noise produced by other devices in the 2.4GHz ISM band, and it amounts to −72.52dBm in average.

Finally, with these data and the equation specified in the 'IEEE Std 802.15.4-2003' reference document [15] the bit error rate (BER) of IEEE 802.15.4 transmissions can be estimated as shown in (2).

$$BER = \frac{8}{15} * \frac{1}{16} * \sum_{k=2}^{16} -1^k * \binom{16}{k} * e^{20 * SINR * \left(\frac{1}{k} - 1\right)} \quad (2)$$

If the message length of the sensing application is taken into account (55 bytes or 440bits, including headers), the packet error rate (PER) can be calculated as shown in (3).

$$PER = 1 - (1 - BER)^L \quad (3)$$

Since the BER of the proposed scenario has a very low value, the probability of two or more simultaneous bit errors in the same frame is negligible, therefore the upper limit of the PER can be estimated as shown in (4).

$$PER \leq L * BER \quad (4)$$

Where L represents in both equations the length of the data frame in bits.

From Fig. 14 and the previous equations one can conclude that with a distance between transmitter and receiver of about 20m the PER is adequately low (around 1.82 * 10^{-5}).

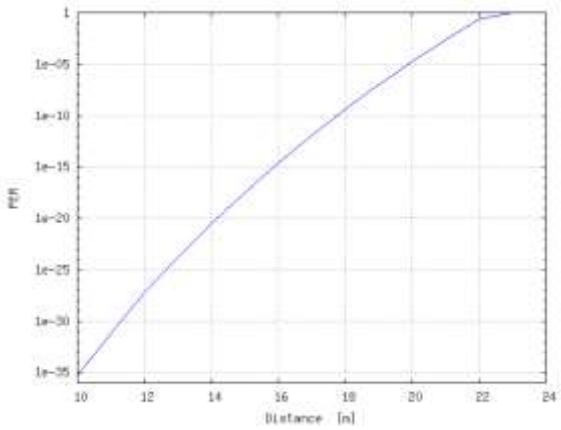

Figure 14. PER as a function of the distance between transmitter and receiver

B. Network topology

Network topology is analyzed in this section in order to be able to use it as a parameter for the battery life calculations. At any instant in time, the routing from the source nodes to the central server (sink) can be seen as a tree graph where leaves are sensing nodes and the root is the gateway device. In this scenario, there are only two parameters needed to characterize the network topology: the maximum number of neighbors of a node *n*, and the maximum number of hops between source and destination *j*.

Fig. 15 shows the basic topology used in the analytical study. In the worst case scenario, a node will route packets coming from *n*-1 of its neighbors (all but the next hop in the route). If this same rule is applied to all the nodes in the network, the result is a tree where all traffic is eventually forwarded by only one node, the one that is closer to the sink. This node will have to forward the traffic from the N previous nodes, where N is defined by (5).

Figure 15. Basic topology with *j*=3 hops and *n*=5 neigbours

$$N = \sum_{i=1}^{j-1} (n-1)^i \quad (5)$$

Taking into account the PER and the maximum retransmission number, for each sampling period there is a total number of N' messages that will be forwarded by the root node, where N' is defined in (6) and p is the probability of a message error.

$$N' = \sum_{i=1}^{j-1} (n-1)^i (1-p^9)^i \quad (6)$$

Also, the expected number of retransmissions of any message (forwarded or not) that any node will have to perform will be denoted as R and is defined in (7).

$$R = 8p^8 + \sum_{i=0}^{7} p^i (1-p) i \quad (7)$$

C. *Battery life*

Taking into account the previous considerations, a set of calculation tables has been built in order to estimate expected battery life of the nodes in several possible scenarios with a variation in the network topology ranging from *j*=2 and *n*=2 (a linearly connected network of only 3 nodes) to *j*=5 and *n*=5, with every possible combination in between.

Table I summarizes the estimated number of messages that will have to be forwarded by the root node per sample interval with the possible combinations of *n* and *j*. Since the PER has been measured to be a very low value of $1.82*10^{-5}$, the number of forwarded messages is approximately equal to the number of preceding nodes of the root.

TABLE I. ESTIMATION OF PACKETS FORWARDED

		# of hops			
		2	***3***	***4***	***5***
# of neighbours	***2***	1	2	3	4
	3	2	6	14	30
	4	3	12	39	120
	5	4	20	84	340

Also the energy draw of every task performed by the node as previously described is taken into account, including its repetition period, its current draw and its duty cycle. The tasks considered in the calculations are:

- **Sleep + base timer**: joint current draw of the microprocessor and TinyOS main timer.
- **ADC**: current draw of the MTS400CA sensor board.
- **Medium sense**: current draw due to the radio preamble sampling task.
- **Battery sensor**: current drawn for each sample taken by the battery level sensor.
- **Sensiron Temperature/Humidity Sensor**: current drawn for each sample taken by the Sensirion sensor.
- **Intersema Temperature/Pressure Sensor**: current drawn for each sample taken by the Intersema sensor.
- **TAOS Light Sensor**: current drawn for each sample taken by the TAOS sensor.
- **Tx-RTE**: Current drawn by each route update message transmission.
- **Rx-RTE**: Current drawn by each route update message reception.
- **Tx-DATA**: Current drawn by each sampling message transmission.
- **Tx-FWD**: Current drawn by each forwarded message transmission.
- **Rx-FWD**: Current drawn by each forwarded message reception.
- **Tx-ERR**: Current drawn due to the retransmissions of failed messages.
- **Rx-DATA**: Current drawn due to messages that are sent by neighbors but are not destined to the current node.

As an example, Table II summarizes the results of these data for a node with *j*=4, *n*=3, a PER of $1.82*10^{-5}$, and a sampling period of 300s.

TABLE II. CURRENT DRAW SOURCES FOR A NODE

	Current (mA)	Duration (ms)	Period (ms)	Duty cycle (%)	Energy (mAh)
Sleep + base timer	0.0428	-	-	100	0.042
ADC	8	0.3	48	0.625	0.05
Medium sense	7	1.8	122	1.4754	0.1032
	25	1	122	0.8196	0.2049
	29	0.44	122	0.3606	0.1045
Battery sensor	8.8	4.7	300000	0.0015	0.0001
	1.1	325	300000	0.1083	0.0011
Sensiron Temp./Hum. Sensor	8.8	8.5	300000	0.0028	0.0002
	0.5	100	300000	0.0333	0.0001
	1	250	300000	0.0833	0.0008
Intersema Temp./Press. Sensor	8.8	13.5	300000	0.0045	0.0003
	7	26.63	300000	0.0088	0.0006
TAOS Light Sensor	8.8	10.5	300000	0.0035	0.0003
	0.6	835	300000	0.2783	0.0016
	8.8	4.5	300000	0.0015	0.0001
Tx-RTE	23	145	600000	0.0241	0.0055
Rx-RTE	23	145	200000	0.0725	0.0137
Tx-DATA	23	145	300000	0.0483	0.0111
Tx-FWD	23	145	21428.5714	0.6766	0.1556
Rx-FWD	19	145	21428.5714	0.6766	0.1285
Tx-ERR	23	145	109888e4	0	0
Rx-DATA	19	145	300000	0.04833	0.0091

With this information, the estimated energy consumption of the root node is as shown in Table III.

TABLE III. ENERGY DRAW AS A FUNCTION OF NETWORK PARAMETERS (mAh)

		# of hops			
		2	***3***	***4***	***5***
# of neighbours	***2***	0.5658	0.5861	0.6064	0.6267
	3	0.5907	0.6719	0.8343	1.1591
	4	0.6156	0.7983	1.3464	2.9907
	5	0.6405	0.9653	2.2645	7.4613

Finally, assuming a defined battery model with a nominal capacity of 20500mAh and a nominal self-discharge rate of 3% a year (industry standard for type D alkaline batteries), the expected battery life in years of the root node in the network (worst case) is as shown in Table IV.

The results show that in networks with a moderate number of nodes and with an environment sampling interval that is more than enough for environmental monitoring applications, the battery life expectancy can be over 2 years. However, as the number of nodes in the network increase, the number of forwarded messages also increases and the root node's battery is rapidly depleted.

TABLE IV. EXPECTED BATTERY LIFE AS A FUNCTION OF NETWORK PARAMETERS (YEARS)

		# of hops			
		2	***3***	***4***	***5***
# of neighbours	***2***	3.6226	3.5143	3.4121	3.3156
	3	3.4907	3.1188	2.5688	1.8966
	4	3.3678	2.6735	1.6474	0.7641
	5	3.2531	2.2479	1.0013	0.3106

VI. Conclusions and future work

A detailed analysis of the expected battery life of the nodes in an environmental monitoring WSN composed of MicaZ motes has been performed. The current draw of the processor, sensors, radio message sending and forwarding have been taken into account in the calculations.

The energy draw has been estimated for network topologies from 2 up to 5 hops and from 2 up to 5 neighbors per node, obtaining battery durations that are well over 2 years for small networks. As the number of nodes in the network increase, the main source of energy consumption turns out to be message forwarding in the nodes closer to the data sink.

Although the calculations are made for the worst case scenario in which only one node forwards the messages of all the rest of the devices, for larger networks the creation of clusters of nodes for routing in order to limit the number of messages forwarded is recommended, so that battery life can be further extended.

Other additional measures that can be taken in order to maximize battery life in this type of scenarios include modifying the sensing application in order to send only data when the values are modified or grouping messages in the forwarding nodes so that data traffic is greatly reduced.

References

[1] F. Ingelrest, G. Barrenetxea, G. Schaefer, M. Vetterli, O. Couach, and M. Parlange, "SensorScope: application-specific sensor network for environmental monitoring," *ACM Transactions on Sensor Networks*, vol. 6, no. 2, article 17, 2010.

[2] N. Xu, S. Rangwala, K. K. Chintalapudi et al., "A wireless sensor network for structural monitoring," in *Proceedings of the Second International Conference on Embedded Networked Sensor Systems (SenSys'04)*, pp. 13–24, ACM Press, New York, NY, USA, November 2004.

[3] Z. Chaczko, A. Kale, and C. Chiu, "Intelligent health care—a motion analysis system for health practitioners," in *Proceedings of the 6th International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pp. 303–308, Brisbane, Australia, December 2010.

[4] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[5] Y. Liu, Y. He, M. Li et al., "Does wireless sensor network scale? A measurement study on GreenOrbs," in Proceedings of the IEEE

International Conference on Computer Communications (INFOCOM '12), pp. 873–881, April 2011.

[6] Ali, Nurul Amirah, Micheal Drieberg, and Patrick Sebastian. "Deployment of MICAz Mote for wireless sensor network applications." Computer Applications and Industrial Electronics (ICCAIE), 2011 IEEE International Conference on. IEEE, 2011.

[7] Navarro, Miguel, et al. "ASWP: a long-term WSN deployment for environmental monitoring." Proceedings of the 12th international conference on Information processing in sensor networks. ACM, 2013.

[8] Paek, Jeongyeup, et al. "A wireless sensor network for structural health monitoring: Performance and experience." Center for Embedded Network Sensing (2005).

[9] Kumar, Vinay, and S. Tiwari. "Performance of routing protocols for beacon-enabled IEEE 802.15. 4 WSNs with different duty cycle." Devices and Communications (ICDeCom), 2011 International Conference on. IEEE, 2011.

[10] Davis, Tyler W., et al. "An experimental study of wsn power efficiency: Micaz networks with XMesh." International Journal of Distributed Sensor Networks 2012 (2012).

[11] Teo, Amos, Gurminder Singh, and John C. McEachen. "Evaluation of the XMesh routing protocol in wireless sensor networks." Circuits and Systems, 2006. MWSCAS'06. 49th IEEE International Midwest Symposium on. Vol. 2. IEEE, 2006.

[12] Su, Weilian, and Mohamad Alzaghal. "Channel propagation characteristics of wireless MICAz sensor nodes." Ad Hoc Networks 7.6 (2009): 1183-1193.

[13] Memsic Inc. "MicaZ Wireless Measurement System." Available online: http://www.memsic.com/userfiles/files/Datasheets/WSN/micaz_datasheet-t.pdf

[14] "TinyOS: An OS for embedded wireless devices." Available Online: https://github.com/tinyos/tinyos-main

[15] "IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks Specific Requirements Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)," in IEEE Std 802.15.4-2003, pp.0_1-670, 2003.

About Author (s):

David Chaves-Diéguez received the M.Sc. degree in telecommunication engineering from the University of Vigo, Vigo, Spain, in 2007, where he is currently pursuing the Ph.D. degree from the Information Technologies Group. He is currently a Researcher with the Galician Research and Development Center in Advanced Telecommunications, Vigo. His current research interests include vehicular and delay/disruption tolerant communication technologies, low-power wireless sensor networks, e-Health systems, and nomadic information systems. He has involved in several projects regarding the above technologies and several contributions to academic conferences and journals in the above fields

Felipe Gil-Castiñeira is an associate professor in the Department of Telematic Engineering at the University of Vigo, Spain. He is also with Gradiant, Spain, as responsible of the Intelligent Networked Systems area. His research interests include wireless communication technologies, embedded systems, and ubiquitous computing. Felipe Gil-Castiñeira received a PhD in telecommunication engineering from University of Vigo.