

SCSP : an Energy Efficient Network-MAC Cross-layer design for wireless Sensor Networks

Bilel Nefzi, Hugo Cruz Sanchez, Ye-Qiong Song

► **To cite this version:**

Bilel Nefzi, Hugo Cruz Sanchez, Ye-Qiong Song. SCSP : an Energy Efficient Network-MAC Cross-layer design for wireless Sensor Networks. The Ninth IEEE International Workshop on Wireless Local Networks - WLN'09, Oct 2009, Zurich, Switzerland. 8 p., 2009. <inria-00431125>

HAL Id: inria-00431125

<https://hal.inria.fr/inria-00431125>

Submitted on 19 Sep 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

SCSP : an Energy Efficient Network-MAC Cross-layer design for wireless Sensor Networks

Bilel Nefzi

LORIA - Nancy University INPL
Campus Scientifique - BP 239 - 54506
Vandoeuvre-Les-Nancy Cedex, France
Email: Bilel.Nefzi@loria.fr

Hugo Cruz-Sanchez

LORIA - Nancy University INPL
Campus Scientifique - BP 239 - 54506
Vandoeuvre-Les-Nancy Cedex, France
Email: Hugo.Cruzsanchez@loria.fr

Ye-Qiong Song

LORIA - Nancy University INPL
Campus Scientifique - BP 239 - 54506
Vandoeuvre-Les-Nancy Cedex, France
Email: Ye-Qiong.Song@loria.fr

Abstract—This article presents SCSP (for “Sleep Collect and Send Protocol”) for wireless sensor networks, a network-MAC cross layer design that resolves the inherent conflict between energy efficiency and throughput. The protocol uses in its MAC layer a new paradigm that we call “sleep, collect and send”. The idea of SCSP is that a router sleeps for a given amount of time, wakes up and collects data from its children and other routers and then send them into a burst during a period of time that we call transmission period. In its network layer, the protocol uses a hierarchical tree structure as network architecture and a tree routing protocol. SCSP does not require synchronization between routers and dynamically calculates the sleep and collect periods according to the amount of incoming traffic. The protocol is implemented and simulated in OPNET simulator.

I. INTRODUCTION

Many wireless sensor networks (WSN) applications such as forest surveillance are deployed for long periods with only batteries as main power supply. In order to take account of energy efficiency, MAC protocol designs use duty cycling mechanisms in which nodes periodically switch between sleep (low-power) and active modes. However, adding sleep periods degrade network throughput which is an essential parameter for some applications and situations. For example, when a fire is detected in the forest, a big number of messages will be generated thus stressing the high throughput functionality of the MAC protocol. Therefore, nodes in WSN must be capable of achieving high throughput rates while minimizing energy consumption. When an urgent event occurs, there is no need for energy saving anymore. Hence, we need a dynamic adaptation mechanism of sleep and active periods.

To address this challenge, we propose the SCSP protocol, a network-MAC cross-layer design that dynamically adapts the active and sleep periods according to the amount of received traffic. SCSP consists of a MAC protocol that works according to the paradigm “sleep, collect and send”, similar to the “store and forward” in wired switches except that we add a sleep state. Besides, it does not require synchronization between routers. Thus, it is scalable. The network layer uses a modified version of the ZigBee [1] tree routing that is more tolerant to node failures. MAC layer provides the list of neighbor routers to the network layer which in its turn provides multiple forwarding choices to it. Hence, MAC protocol has the possibility to change the next-hop router during transmission

if the message is not successfully transmitted after some tries.

The rest of the paper is organized as follows. Section II presents the related work. In section III we describe the SCSP design. In section IV, we evaluate its performance through the simulation of a tunnel supervision scenario. Section V concludes the paper.

II. RELATED WORK

S-MAC [2] and B-MAC [4] are the most popular contention based MAC protocols in WSN. S-MAC is designed with energy consumption minimization and self-configuration support in mind. S-MAC periodically switches between sleep and active states. It provides a synchronization mechanism that maintains a common sleep schedules between neighbor nodes which enables low-duty-cycle operation in multi-hop networks while minimizing energy consumption. The protocol uses an RTS/CTS transmission scheme and a virtual carrier sense technique similar to those in IEEE 802.11 [3] which are used to provide adaptive listening and avoid overhearing unnecessary traffic. Finally, it presents a message passing technique which fragments a long message into many small fragments and transmit them in a burst.

B-MAC provides power management via low power listening. Nodes periodically switch between sleep and active states (sleep period is called check interval). When a node wakes up, it checks channel activity. If activity is detected, the node stays awake and receives the incoming packet. However, if the medium is clear, the node goes back to sleep. Each transmission is preceded by a preamble as long as the check interval so the intended receiver is definitely aware of the transmission and receives the incoming packet. SCSP has the advantages of both protocols and provides further enhancements. It uses the sleep and wake up periods as S-MAC does. However, during the sleep periods, a node periodically wakes up, checks channel activity and stays active if the channel is not free as B-MAC does. This combination avoids S-MAC’s synchronization overhead thus improving scalability. In addition, by bursting data, a preamble is sent once for the first message to transmit thus improving throughput. Finally, SCSP dynamically adapts active and sleep periods while they have to be set during network deployment or through reconfiguration mechanisms for S-MAC and B-MAC.

IEEE 802.15.4[5] standard specifies the physical layer (PHY) and MAC sublayer for low data rate, low-cost wireless personal area networks (WPANs) of fixed, portable or moving devices with no battery or very limited energy consumption requirements. In its beacon enabled mode, the MAC sublayer uses a superframe structure, defined as the time between two successive beacon transmissions, that is composed of an active period and a sleep period. The superframe structure is adapted to a star topology. [6] proposed a solution to enable the use of superframe structures in multi-hop networks. However, it is synchronized and centralized thus non scalable.

Reactive routing protocols like AODV [7] and DSR [8] and pro-active routing protocols like OLSR [9] and DSDV [10] are not suitable for duty cycled MAC protocols. Path discovery in reactive protocols has a high delay and energy costs while in pro-active protocols, performing frequent route updates and taking sleep and wake up schedules into account is expensive. In our protocol we used a modified ZigBee tree routing (ZTR and m-ZTR) scheme which does not require any message exchange or route discovery. The modification tackles some of its weakness, namely its tolerance to node failure.

III. SCSP DESIGN

A. General parameters and network architecture

We consider a wireless network composed of simple nodes and router nodes (named router in the rest of the document). A simple node is a device that generally has measurement sensors built in and has limited routing decisions capabilities. A router is a device that implements full routing and network management protocols. However, it can act like a simple device when its routing and management capabilities are disabled.

The network is organized into a tree and uses the tree addressing scheme presented in ZigBee standard. The network formation procedure will be developed in Sect. III-C.

All nodes (simple nodes and routers) use CSMA/CA protocol to access the medium but with different maximum backoff intervals in order to give routers higher chance to access the medium; we used the unslotted version of CSMA/CA proposed by IEEE 802.15.4 (used in the non beacon enabled mode). SCSP operates on the top of the "2450 MHz DSSS IEEE 805.15.4" physical layer; the data rate is equal to 250 kbps.

B. Channel access protocol

1) *General description:* When a node starts, it executes the association procedure to integrate the network. For a router, once the association is successfully completed, it starts sending beacon frames to provide useful information to its neighbors (including its children) and synchronization to its children. Routers use a superframe structure, defined as the time between two successive beacon frames. The superframe is divided into three periods; Sleep Period (SP), Waiting Period (WP) and Transmission Period (TP). We call the sum of SP and WP a subframe. During the SP, the router enters into sleep mode. When the SP expires, the router becomes active.

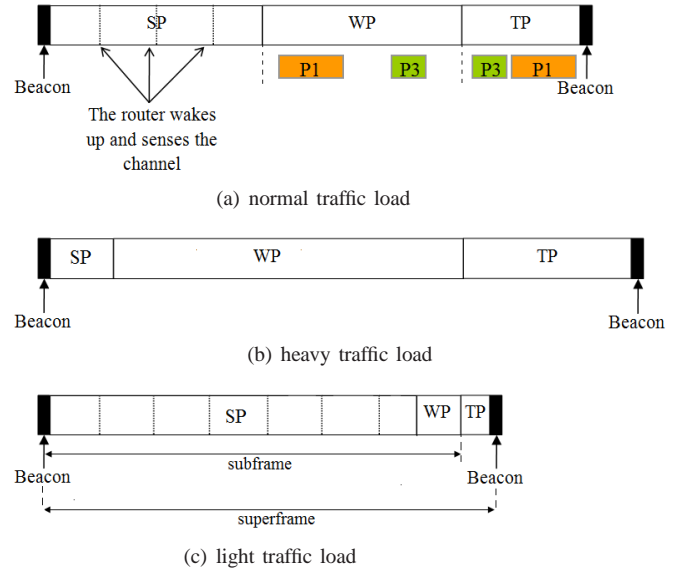


Fig. 1. Subframe structure

However it does not transmit packets one by one upon their arrival. Instead, it goes to reception state and collects data from its neighbors (simple nodes or routers) during the WP. After the end of the WP, the router starts transmitting all packets queued in its buffer in a single burst. This is done in the TP. Finally, at the end of the TP, the router sends the next beacon and starts another cycle. The superframe structure is illustrated in Fig. 1(a). Simple nodes are always in sleep mode unless they receive a packet to transmit from the higher layer. Full details of the WP, TP, SP and the operating mechanism of simple devices are given next.

2) *Waiting period:* The WP depends on the amount of incoming traffic; the higher the traffic volume, the longer the WP duration. The WP duration for a router r is given by (1).

$$WP = \begin{cases} N_{max} \cdot d_S, & \text{if } N_c^r \neq 0 \\ N_{max} \cdot d_R, & \text{otherwise} \end{cases} \quad (1)$$

N_c^r is the number of children associated to the router r .

d_S and d_R are the maximum amount of time needed by a simple node and a router, respectively, to perform the collision avoidance procedure (d_{BP}), sense the channel (t_{CCA}) and detect that it is Idle (at the first attempt), change the antenna state from receiver to sender (t_{tat}), send the packet (d_{pkt}), wait for an acknowledgment (t_{ack}), and receive it (d_{ack}). See Fig. 2. d_S or d_R define one slot time.

N_{max} is an integer that has to be estimated using the incoming traffic (it defines the number of slots). Its value gives the duration of the WP. It has a minimum value of 1. The estimation algorithm is described next.

The estimation algorithm. Let's define first the set $\Omega \in \mathbb{N}$ as the set of WPs during which the router receives data and WP_k , $k \in \Omega$ these WPs. We define also, U_k , $k \in \Omega$ as the ratio giving the sum of the service times of received packets ($\sum_{i \in E_k} \frac{1}{\mu_i}$, E_k is the set of received packets during WP_k) divided by WP_k duration. $\frac{1}{\mu_i}$ is the service time of packet

i (see Fig. 2). U_k is shown in (2). We note that U_k can be greater than 1 in the case where the router is still receiving a packet from a neighbor router while the WP ends (recall that the router is not preemptive).

$$U_k = \frac{\sum_{i \in E_k} \frac{1}{\mu_i}}{WP_k} \quad (2)$$

Let us denote by S_k the exponential moving average of U_k , $k \in \Omega$

$$S_k = (1 - \alpha)S_{k-1} + \alpha U_{k-1} \quad (3)$$

α is the smoothing factor. We used a non linear filter where α is bigger when $U_{k-1} \geq S_{k-1}$ allowing S_k to adapt more swiftly to traffic increase. After a number of trials, we found that values of α , $\alpha_1 = 0.008$ and $\alpha_2 = 0.01$, gave the best of all-around performance. The computation of S_k resembles that of the average queue length in RED [13] and the round trip time in TCP [14].

S_k is evaluated before sending the k^{th} beacon frame and is used in the Algorithm 1 to determine the value of N_{max} for the k^{th} WP (thus its duration). In the algorithm, Thr_{max} , Thr_{min} and $NMAX$ designate the maximum threshold of S_k , the minimum threshold of S_k and the maximum value of N_{max} , respectively. Thr_{max} allows the increase of the WP 's duration. When set too high, the WP will not respond rapidly to the network load growth. When set too low, bandwidth will be wasted since the WP will increase too fast. Thr_{min} is the minimum utilization ratio under which the protocol performance will degrade significantly. Thr_{min} is used to decrease the WP 's duration : set too high, WP decreases fast which diminish the benefits of our algorithm. When set too low, WP 's duration decreases too slowly compared with the network load growth. $NMAX$ bounds the value of N_{max} to prevent a high increase in the WP duration. It is set to 15 in all simulations.

Intuitively, S_k calculates the average utilization ratio of the WP . If S_k exceeds Thr_{max} , the value of the N_{max} is increased which augmentates the duration of the WP . If S_k decreases below Thr_{min} , the value of N_{max} is decreased. We can see that changing the WP duration, affects the value of U_k (2) and so S_k which in its turn affects the value of N_{max} . Besides, S_k depends also on the traffic load. Hence, our algorithm appears like an iterative process where the objective is to find the good value of N_{max} that reflects the incoming traffic load while satisfying an utilization ratio between Thr_{min} and Thr_{max} .

3) *Transmission period*: The collected data during the waiting period are transmitted in a burst. At the beginning of the TP, the router sends a preamble frame; its duration is superior to the wake up period of routers¹. This way, all neighbor routers will wake up. However, if the network is configured to stay active all the time, preamble transmission is disabled. After that it sends the queued packets. CSMA/CA protocol is used to transmit the preamble packet. Then, the remaining

¹The wake up period is the same for all routers

Algorithm 1: Estimation algorithm

```

/* Update  $S_k$  */
 $\alpha = \alpha_1$ 
if  $U_{k-1} \geq S_{k-1}$  then
  |  $\alpha = \alpha_2$ 
end
 $S_k = (1 - \alpha)S_{k-1} + \alpha U_{k-1}$ 
/* Update  $N_{max}$  */
if ( $S_k \geq Thr_{max}$ ) then
  |  $N_{max} = N_{max} + 1$ 
else
  | if  $S_k \leq Thr_{min}$  then
    |  $N_{max} = N_{max} - 1$ 
  | end
end
 $N_{max} = \max(1, \min(N_{max}, NMAX))$ 

```

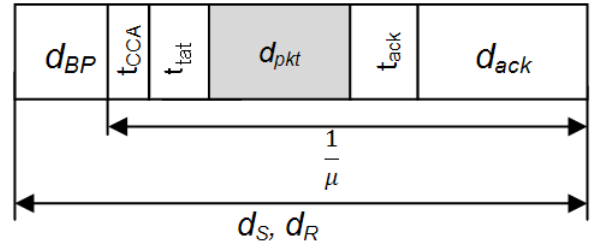


Fig. 2. d_S , d_R and $\frac{1}{\mu}$ variables

packets are transmitted directly upon the reception of the acknowledgement², except for the first queued packet (because preamble packets are not acknowledged). If the sender receives an acknowledgement for a transmitted packet, it drops the packet and moves to the next one in the buffer. If not, it reuses CSMA/CA to send the non successfully transmitted packet. At the end of the TP, the router sends a beacon frame and goes to sleep.

Any router has at least one message to send during the TP which is the beacon frame. If the router has buffered messages to send, it transmits the beacon after finishing the transmission of these buffered messages. Otherwise, it must contend to access the channel using CSMA/CA before sending the beacon frame. In this case, the preamble frame is not sent. We shall notice that a beacon frame transmission is always considered successful since beacons are not acknowledged.

After the end of the WP , the transmission procedure described above begins only if the channel is idle. Otherwise, the WP is extended until it becomes idle. At each detection of a busy state, the WP is extended by d_S seconds (d_S is defined in Sect. III-B2).

4) *Sleep period*: During the SP, the router goes to sleep. However it periodically wakes up and senses the channel. If the channel is free it goes back again to sleep, otherwise it remains

²Acknowledgments are enabled in SCSP

active as far as it receives data from the medium. The router goes back to sleep if the SP did not expire yet (in that case, it starts the WP) and if the channel remains idle for a period of time equal to the sum of d_S and the maximum backoff duration of a router. This amount of time guarantees that no other router is starting its transmission procedure before it goes to sleep.

The duration of the subframe is a constant defined by the network administrator and the duration of the WP depends on the traffic load. Thus, the duration of the SP depends also on the traffic. When the traffic load increases, the WP increases while the SP decreases and vice-versa. This is the dynamic adaptation of the sleep and active periods which is illustrated in Fig. 1(b) and Fig. 1(c) for a heavy and a light traffic, respectively.

5) *Simple nodes operating mechanism*: A simple node is always in sleep mode unless it receives a packet to transmit from the higher layer. In that case, it wakes up and listens to the channel. It keeps listening to the channel until it receives a beacon frame from a neighbor router. Then, it extracts the SP and WP durations and the address of the router which sends this beacon and goes back to sleep during the SP period specified in this beacon frame. After the expiration of SP, it wakes up again. Now the simple node is synchronized with the WP of the beacon frame sender. Finally, it sets the next hop address of the packet to the address extracted from the beacon frame and uses CSMA/CA to send it.

If a node has another packet to transmit or if the transmission of the packet being transmitted fails (due to a CSMA/CA failure or a non reception of an acknowledgment frame), the simple node must check first if it can transmit it in the remaining time of the WP. If it is the case, it sets the next hop address of the packet to the address extracted from the beacon frame and uses CSMA/CA to send it as the previous packet. Otherwise, it must restart the whole procedure; locate a beacon frame, sleep during the SP period of the beacon frame of the sender and send the packet.

Any other packet received during the transmission procedure is queued. At the end of transmission of all the queued packets, the simple node goes back to sleep. Note that the next hop can be a router that is not the father of the simple node.

C. Network formation

1) *Association*: We denote by the root, the router that starts the network and specifies all its parameters (like the subframe length and parameters needed for address attribution). The root is equivalent to the ZigBee coordinator in the ZigBee standard.

The association mechanism works as follows. When the root becomes active, it starts sending beacon frames periodically; the root starts using the superframe structure immediately. If a router or a simple device wants to join the network, it starts collecting beacons during a period of time that we call "collection period" (CP). During this period, for any beacon received, the node records all the information provided and stores the record into a list. The list is sorted according to the

depth of the sender which is the minimum number of hops to reach the root using only parent child links and the number of children associated to it; the head of the list contains the record with the minimum depth and the number of children. When the CP expires, if the list is empty, the CP restarts. Otherwise, the node selects the head of the list and tries to locate the next beacon of the "selected node", in order to be synchronized with it. Finally, it sends an association request during WP of the selected sender. Upon the reception of an association request, a router sends immediately a reception response containing the assigned address and the network parameters. This is the only case where a given router sends a frame during the WP. The association response frame acts like an acknowledgment; if a node sends an association request and does not receive an association response, it considers the frame lost and moves to the next record in the list if any and repeats the synchronization process. Otherwise, it starts again the CP. The association procedure is executed once for all routers, during network establishment, unless the root decides to reset the network.

2) *Address attribution*: The network addresses are assigned using a distributed address allocation scheme that is designed to provide every potential parent with a finite sub-block of network addresses to distribute to its children. During network establishment, the root determines the maximum number of children per parent (C_m), the maximum number of routers (R_m) between these children as well as the maximum depth of the network (L_m) (the root has a depth of 0). A function called $Cskip(d)$ (equation 4) is used after that to calculate the size of the address sub-bloc being distributed by each parent located at depth d .

$$Cskip(d) = \begin{cases} 1 + C_m \cdot (L_m - d - 1), & \text{if } R_m=1 \\ \frac{1 + C_m - R_m - C_m \cdot R_m^{L_m - d - 1}}{1 - R_m}, & \text{otherwise} \end{cases} \quad (4)$$

The network address distribution is as follows. The root has always the address 0. For routers, the address assignment uses the value of $Cskip(d)$ as an offset: if the node is the first served, its address is one greater than its parent address. Otherwise, the addresses are separated from each other by $Cskip(d)$. For simple nodes, network addresses are assigned in a sequential manner using the following rule :

$$A_n = A_{parent} + Cskip(d) \cdot R_m + n \quad (5)$$

Here $1 \leq n \leq (C_m - R_m)$ and A_{parent} represents the address of the parent.

D. Routing protocol

1) *ZigBee tree routing*: End devices (simple nodes in our case) always send their data to their father. For routers, ZigBee tree routing (ZTR) is based on some comparison rules: for a ZigBee Router with address A at depth d , if the following logical expression is true, then a destination device with address D is a descendant.

$$A < D < A + Cskip(d - 1) \quad (6)$$

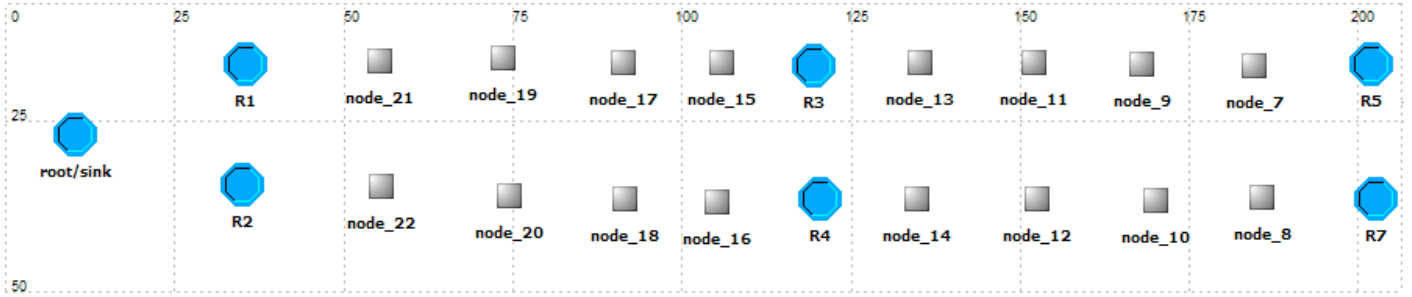


Fig. 3. Simulation scenario

If it is the case, the Next Hop is given by the following rule:

- $N = D$ for ZigBee end devices where $D > A + Rm \cdot Cskip(d)$ (7)
- $N = A + 1 + \lfloor \frac{D-(A+1)}{Cskip(d)} \rfloor \times Cskip(d)$, otherwise

If the expression is not true, the destination is not a descendant and the message should be routed through A 's parent.

2) *Modifications*: First, simple devices does not always send their data to their father. Instead, they send it to the first received beacon sender. Finally, in prior work, Nefzi [11] proposed an improvement of ZTR using neighborhood table in routing decision in order to improve the worse-case delay. Our modification follows the same logic since neighborhood table can be efficiently built using beacon information. The modification works as follows. MAC layer provides the list of neighbor routers to the network layer. This table is updated whenever a router receives a beacon frame. First, by default, the next-hop address is the one provided by ZTR with only one exception; if the destination is a descendant of a neighbor node, the next-hop is set to this neighbor. Finally, if the next-hop as calculated by ZTR is not a descendant, the network layer sorts the neighbor table according routers depth; the neighbor router with the minimum depth is placed first.

During TP, if a transmission fails after some tries and if the next hop is the father and the destination is not the father (MAC layer knows its father address), it picks up the head of the list and sends the message again to this router. If the neighbor table is empty or the destination is a descendant, the message is destroyed.

The modification works when the next-hop as given by ZTR is the father. This is because the sink node is generally the root. Besides, first level nodes tend to die first because most of the traffic crosses them to reach the destination.

IV. SIMULATION ANALYSIS

We present in this section a simulation study of SCSP which we implemented in OPNET [15] (Modeler v14.5). We analyze the energy consumption and end to end delay with different network conditions and protocol parameters. We study its dynamic throughput adaptation behavior. We compare m-ZTR with the basic ZTR. For energy consumption calculation, we use the micaz mote [16] current and voltage parameters which

Receive mode current (mA)	19.7
TX current (mA)	17.4
Active mode current(mA)	8
Sleep mode current (μ A)	15
Voltage (V)	3

TABLE I
MICAZ MOTE PARAMETERS

Data rate (kb/s)	250
CCA duration (μ s)	0.128
Initial energy (J)	100
Wake up duration (ms)	10
Thr_{max}	0.75
Thr_{min}	0.28
d_S (ms)	4.816
d_R (ms)	3.536

TABLE II
SIMULATION PARAMETERS

are given in table I. Table II summarizes general simulation parameters³.

A. Scenario description

We consider a multi-hop network, which simulates a tunnel supervision system, in a surface of 250m x 50 m (Fig. 3). It is composed of seven routers (blue circles in the figure) and 16 simple nodes (gray squares). Simple nodes hear only the two nearest routers and the transmission range of routers is equal to the distance between R3 and R2 (86m). The sink node is the root node. Cm , Rm and Lm are equal to eight, four, and four, respectively. Simple nodes generate a traffic with a constant inter arrival time and a constant packet length equal to 450 bits⁴. The root node is always active and has unlimited energy.

B. Network lifetime

We define the network lifetime as the time of the first router failure. In this set of simulations, we varied the sub-frame

³We set the initial energy to 100J. However, a typical Alkaline AA battery which is used by micaz motes has a theoretical energy of 15390J [17]

⁴We supposed that the deployed application generates periodic messages. However, SCSP is not specifically designed for such applications.

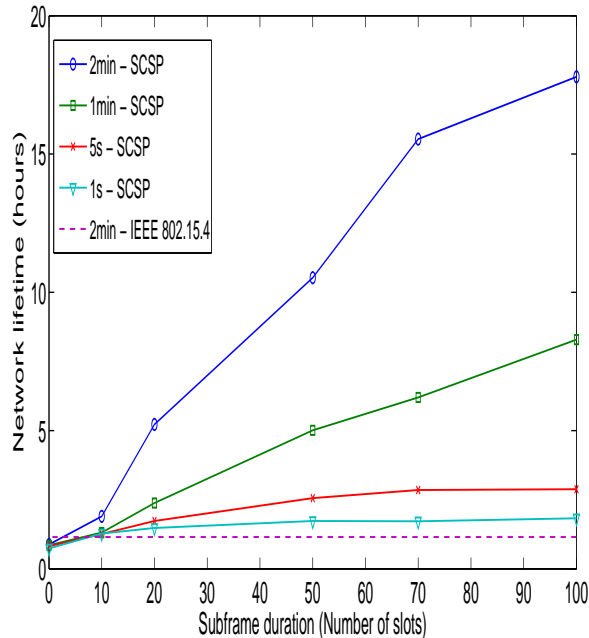


Fig. 4. Energy consumption with different message generation rates

length and then measured the network life time. As WP, the subframe length is measured by the number of slots X_{max} multiplied by d_S or d_R depending on whether the router has children composed by simple nodes. $N_{max} = 0$ means that the node is always active. Results are plotted for 1s, 5s, 1min and 2min generation rates and are illustrated in Fig.4. We measured also the network lifetime of the IEEE 802.15.4 MAC, non beacon enabled mode with static routing, in order to obtain the performance of CSMA/CA in active mode.

The network lifetime increases almost linearly as the sub-frame length increases because routers sleep much more time. For light traffic load (2min and 1min message generation rates), the network lifetime increases as the subframe duration increases. The improvement in comparison with the "always active" scenario is 2 to 20 times depending on the duration of the subframe. For heavier traffic load, network lifetime is almost constant; on one hand, routers wake up more often during their SP to get data and on the other hand, the WP duration is longer since it is adapted according to amount of incoming traffic. SCSP obtains better results except when routers are always active. This is because SCSP has higher MAC overhead than IEEE 802.15.4 non beacon enabled mode; beacon transmissions for instance.

C. End to end delay

We run the previous set of simulations again and measured the average end to end delay as a function of the subframe length. The results illustrated in Fig.5 show that IEEE 802.15.4 MAC always outperforms SCSP. This is because of SP and beacon transmissions overhead. However, if we take into account the inter arrival time characteristic of the traffic,

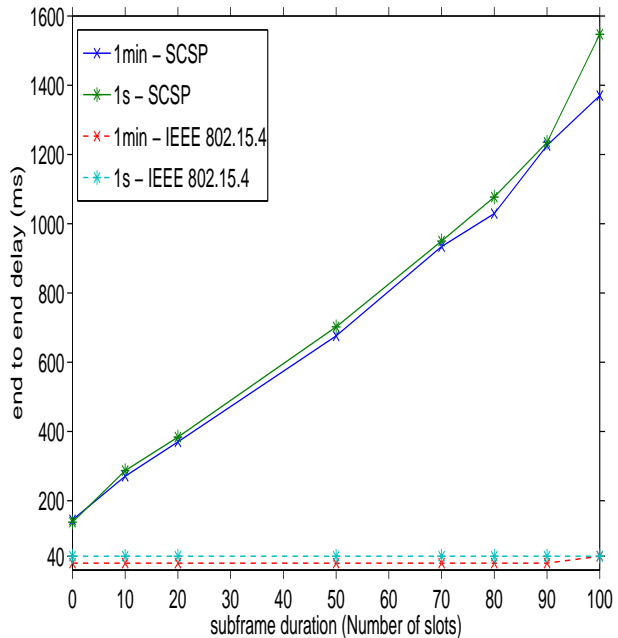


Fig. 5. Comparison of end to end delays between SCSP and IEEE 802.15.4

we can establish a trade off between energy saving and end to end transmission delay. For example, for an inter arrival time equal to one minute and a subframe length equal to 100 slots, the average end to end delay is equal to 1560ms which is considered reasonable in comparison with the 60 seconds of inter arrival time.

Hence, considering the traffic characteristics, the end to end delay and the energy consumption, the network administrator can efficiently calculate the optimal subframe length that maximizes the network lifetime.

D. WP adaptation

In this set of simulations, simple nodes 11 to 18 transmit data starting at 1500s and ending at 2100s with a generation rate of two messages per second. The rest of simple nodes start transmitting at 10s until the end of the simulation with a message generation rate of five seconds. The subframe length is equal to 20 slots. We used the m-ZTR. The energy is set to maximum to avoid nodes death. We plot the WP duration as a function of the simulated time for all routers. Results obtained by routers R2 and R3 are shown in Fig. 6 along with the total traffic sent (in bits/sec) and results of the other routers are illustrated in Fig. 7.

For all routers except R1, the WP duration increases after 10s. This is because some simple nodes start transmitting data at the same time (10s). Thus, the WP is adapted quickly to support this increase in the traffic volume. After that, the WP of all routers goes again to its minimum value, because the generated traffic volume is not big and is smoothed according to the active and sleep router schedules. Similarly, between 1500s and 2100s, eight nodes start transmitting data. Again,

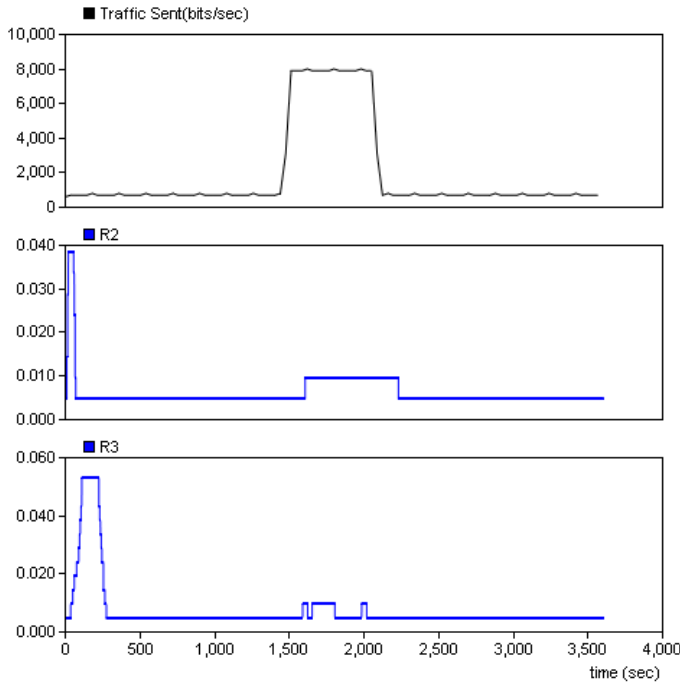


Fig. 6. Total traffic sent in bits/s and WP duration in seconds of R2 and R3

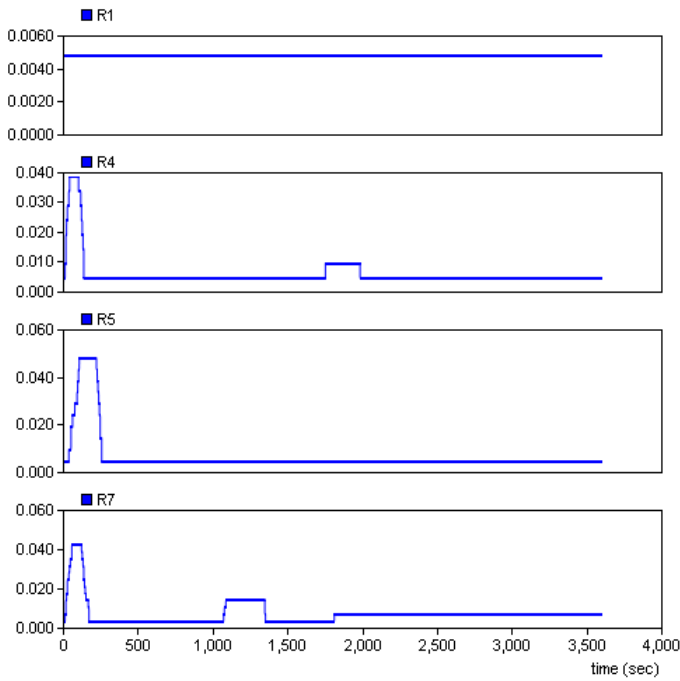


Fig. 7. WP duration in seconds of R1, R4, R5 and R7

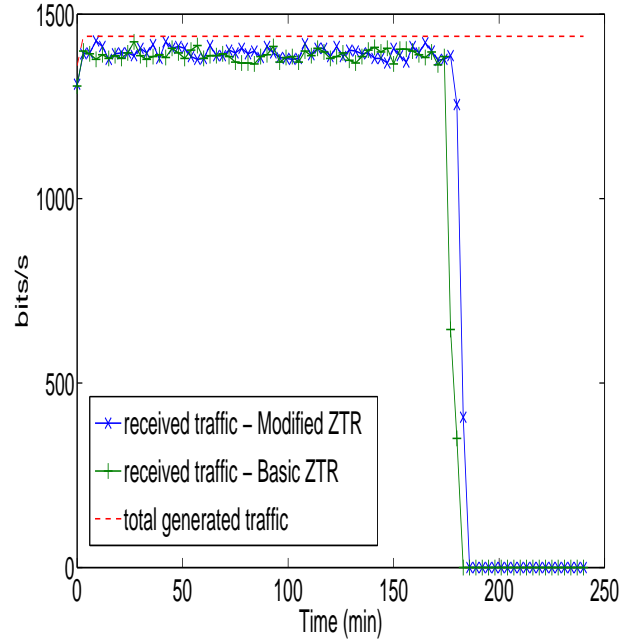


Fig. 8. Comparison between basic and modified ZTR

R2, R3 and R4 adapt their WP to this increase in traffic volume because most of simple nodes that start transmitting at 1500s send traffic to R3 or R4 which have R2 as father. Note that the WP duration of R2 remains equal to 9.632ms (two slots) between 1500s and 2100s because the traffic volume corresponds to a WP duration of two slots. However, the WP duration of R3 and R4 oscillate between one and two time slots. This is because, simple nodes are free to choose the router to which they send data. So, R3 adapts its WP when it is chosen and vice-versa. Since, the traffic generation rate is constant, simple nodes generally wake up at the same time and tend to select the same router. WP of R1 and R5 remains at minimum since most of the traffic is routed through R7 and R2.

SCSP efficiently adapts its WP duration to the incoming traffic volume. This is an important feature of our protocol since it solves the problem of throughput degradation introduced by forcing nodes to periodically enter sleep mode.

E. Routing

In this set of simulations, we fixed the subframe length to 10 slots and data generation rate to one message every five seconds. Then we measured the traffic received by the sink node and the total energy consumption during the simulated time.

Figure 8 shows the performance of ZTR and m-ZTR. Before first router's death, m-ZTR performs slightly better than ZTR due to the additional transmission attempt to a neighbor node when the maximum retransmission limit is reached (it is fixed to five in all simulations). After the first router's death (R1 which is the same for both protocols) and compared to ZTR,

m-ZTR maintains the whole network connectivity for three additional minutes before the received traffic falls due to the death of R2. This is due to the flexibility of selection of next-hop routers by simple nodes and routers.

Hence, m-ZTR is more tolerant to node failures than ZTR.

V. CONCLUSION

This paper presents SCSP, a network-MAC cross-layer design which ameliorates energy consumption by switching between active and sleep periods and dynamically adapting them according to the amount of received traffic. The protocol uses a simple but efficient routing protocol that does not need route maintenance or discovery and works jointly with the MAC layer to enhance its fault tolerance properties. Simulation results show that SCSP, extends the network lifetime and connectivity in comparison with IEEE 802.15.4 MAC.

Future work consists of implementing the protocol within micaz motes.

REFERENCES

- [1] 473-489 ZigBee Specification Document 053474r17. <http://www.zigbee.org>.
- [2] Ye, W., Heidemann, J., and Estrin, D. 2004. Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Trans. Netw.* 12, 3 (Jun. 2004), 493-506. DOI=<http://dx.doi.org/10.1109/TNET.2004.828953>.
- [3] IEEE 802.11b/D3.0, Wireless LAN Medium Access Control (MAC) and Physical (PHY) Layer Specification: High Speed Physical Layer Extensions in the 2.4 GHz Band, 1999.
- [4] Polastre, J., Hill, J., and Culler, D. 2004. Versatile low power media access for wireless sensor networks. In *Proceedings of the 2nd international Conference on Embedded Networked Sensor Systems (Baltimore, MD, USA, November 03 - 05, 2004)*. SenSys '04. ACM, New York, NY, 95-107.
- [5] IEEE-TG15.4 (2003). Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for low-rate Wireless Personal Area Networks (LR-WPANs). IEEE standard for information technology.
- [6] Koubaa, A., Cunha, A., and Alves, M. 2007. A Time Division Beacon Scheduling Mechanism for IEEE 802.15.4/Zigbee Cluster-Tree Wireless Sensor Networks. In *Proceedings of the 19th Euro-micro Conference on Real-Time Systems (July 04 - 06, 2007)*. ECRTS. IEEE Computer Society, Washington, DC, 125-135. DOI=<http://dx.doi.org/10.1109/ECRTS.2007.82>
- [7] Charles E. Perkins, Elizabeth M. Belding-Royer, and Samir Das. "Ad Hoc On Demand Distance Vector (AODV) Routing." IETF RFC 3561.
- [8] David Johnson, David Maltz, Yih-Chun Hu: The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks for IPv4, RFC 4728.
- [9] Philippe Jacquet, Paul Muhlethaler, Amir Qayyum, Anis Laouiti, Laurent viennot, Thomas Clausen. Optimized Link State Routing Protocol (OLSR), RFC 3626.
- [10] C. E. Perkins P. Bhagwat Highly Dynamic Destination-Sequenced Distance Vector (DSDV) for Mobile Computers Proc. of the SIGCOMM 1994 Conference on Communications Architectures, Protocols and Applications, Aug 1994, pp 234-244.
- [11] B. Nefzi, Y.-Q. Song. "Performance Analysis and improvement of ZigBee routing protocol", in: 7th IFAC International Conference on Fieldbuses & Networks in Industrial & Embedded Systems - FeT'2007, Toulouse France, 2007.
- [12] Injong Rhee; Warrior, A.; Aia, M.; Jeongki Min; Sichertiu, M.L., "Z-MAC: A Hybrid MAC for Wireless Sensor Networks," *Networking, IEEE/ACM Transactions on*, vol.16, no.3, pp.511-524, June 2008.
- [13] Floyd, Sally; Jacobson, Van (August 1993). "Random Early Detection (RED) gateways for Congestion Avoidance". *IEEE/ACM Transactions on Networking* 1 (4): 397413. doi:10.1109/90.251892.
- [14] Jon Postel, Transport Control Protocol, RFC 793, September 1981.
- [15] OPNET, "OPNET Simulator, v 14.5, <http://www.opnet.com/>."
- [16] Micaz datasheet, <http://www.xbow.com/>.
- [17] "Alcaline Technical Information". Energizer. Retrieved 11 July 2007.