

Energy Efficient Cross-Layer MAC Protocol for Wireless Sensor Networks

Bouabdellah KECHAR¹, Ahmed LOUAZANI², Larbi SEKHRI¹, Mohamed Faycal KHELFI³
Department of Computer Science, Faculty of Science,
University of Oran Es-Senia, BP. 1524 Oran, Algeria

¹{Kechar.bouabdellah, Larbi.sekhri}@univ-oran.dz, ²choudangm@gmail.com, ³mf_khelfi@yahoo.fr

Abstract

Technological Advances in microelectronic and telecommunication fields enable the existence of tiny computing units, very small and more efficient. In the context of these new technologies, Wireless Sensor Networks (WSN) have emerged. The main research objective in WSN domain is the development of algorithms and protocols ensuring minimal energy consumption. Most proposed solutions are based on one-layer stack model approach. Recently, other works tend to exploit together many layers in order to optimize energy consumption. In this paper, we propose a Cross-Layer Medium Access Control (CL-MAC) protocol using two adjacent layers (MAC and Network) to economize energy for WSN. The basic idea behind our protocol is to wake-up only nodes belonging to a routing path from the source to the base station (Sink) by exploiting routing information while other nodes leave maintained as long time as possible in a sleep mode. The protocol is modeled using a time Petri net and validated by TiNA tool. The simulation results demonstrate the effectiveness of our proposal in terms of energy consumption and latency.

Keywords: Wireless Sensor Networks, Cross-layer optimization, MAC Protocol, Energy consumption, Time Petri net.

1. INTRODUCTION

Wireless Sensor Networks (WSN) are composed by tiny sensors called sensor nodes. A sensor node ensures main functions like acquisition of environmental physical measures, local processing of collected data and their transmission to the Sink. These functions are performed by an electronically modules composing the node's hardware architecture. Other components can be added to ensure mobility function and energy self-recharging. A node is alimented by a limited quantity of energy (battery) and communicates wirelessly using a radio signal. Generally, communication between nodes consumes more energy than local processing or collecting data operation. The geographical nature of the deployment space of nodes makes quasi impossible the replacement or the recharging operations of batteries. The challenge is to economize energy inside every node in order to maintain as long as possible the network functionality. Many research works are developed for energy efficiency at each layer of protocol stack by proposing new algorithms and protocols. In particular, MAC layer was of great interest for many researchers because it is considered as an important source of energy wastage that we summarize as follows [1, 2]:

- Overhearing: a sensor node receives packets that are transmitted for other nodes. This is due essentially to the radio transmission nature (omni-directional) forcing every node of the neighbourhood to waste energy when receiving this radio.
- Collision: since radio channel is shared by many nodes, a collision take place every time when two nodes try to send in the same time their packets. Collisions increase energy consumption and latency in case of packets deliverance mechanism due to retransmissions.
- Control packets (*overhead*): packet headers and control packets (*RTS/CTS/ACK*) used by a MAC protocol do not contain application data, thus they are considered as supplementary data (*overhead*). Control packets can be of importance since most applications use data packets with reduced size.
- Idle listening: when a node is not active, leaves listening the signal carrier to knowing if it is the receiver of an eventually traffic. In this situation, the amount of energy waste is equal to the energy dissipated by a normal reception.
- Over emitting: This case arrives when a sensor node receives a packet while is in a sleep mode. This situation forces the sender to perform new retransmissions that are strongly linked to non-synchronisation problem and therefore consume more energy.

In order to decrease or if possible to eliminate these various sources of energy wastage, several protocols has been proposed these last years and which are divided into two main classes:

1. TDMA-based protocols: these protocols known as deterministic are employed to avoid collisions by associating a slot time for each sensor node in a given cluster, and to mitigate the effects of overhearing problem, because in this situation each node knows his corresponding slot time to transmit data packet [3]. However, these protocols require the presence of a management authority (for example a dedicated access point) to orchestrate the various activities inside a cluster. This makes the use of these protocols more complex in the WSN where the nodes in general have a same priority and very limited resources.
2. Contention-based protocols: these protocols known as CSMA-based are usually used in the multi-hop wireless networking due to their simplicity and their adequacy to be implemented in a decentralized environment like WSN. When these protocols are used, collisions can occur in case of a receiver is located in the radio range of at least two sensor nodes transmitting simultaneously data packets to it. In this situation, this node will not succeed in receiving any data packet. These collisions generate useless retransmissions which cause energy consumption wastage and time consuming in data transmission. To decrease these collisions and to reduce considerably other sources of energy wastage, the Wake-up/Sleep mechanisms and/or the control messages *RTS/CTS/ACK* defined in 802.11x standard, are used to design energy efficient MAC protocols for WSN like S-MAC, T-MAC, B-MAC and Z-MAC.

Recently, a new generation of MAC protocols (*Cross-layer MAC protocols*) using several layers in order to optimize energy consumption has been emerged. These layers can be exploited into two modes: *interaction* or *unification*. In the interaction mode, the MAC protocol is built by exploiting the data generated by other adjacent layers. MAC-CROSS Protocol [4] is an example of Cross-layer approach which allow the routing information of the network layer to be exploited by MAC layer (interaction between MAC and network layers) by leaving only the communicating nodes in activity and by putting into Sleep mode the other neighbour nodes (not concerned by this communication). In order to avoid collisions, MAC-CROSS uses the control messages *RTS/CTS/ACK*. On the other hand, a Cross-layer design mode by unification requires the development of only one layer including at the same time functionalities of considered layers.

In this paper, we propose a Cross-layer protocol named CL-MAC, based on the same ideas used by MAC-CROSS. The fundamental difference between our proposal and MAC-CROSS is at the level of the number of consecutive nodes that are implied in MAC functioning at each frame. Indeed, MAC-CROSS acts on three consecutive nodes while CL-MAC uses all the nodes included in a given routing path from the source to the Sink in one frame, thus making it possible to take place simultaneously with two main operations: data packets transmission between communicating nodes (a process relatively slow) which advances progressively, and control packets transmission which advances quickly towards the Sink, by preparing a path in which only nodes belonging the routing path remain in activity and all other nodes enter into Sleep mode for a given time interval.

Temporal Petri nets are introduced in order to model the proposed protocol and TiNA tool is carried out for validation purpose. A comparative study between CL-MAC and S-MAC in term of energy efficiency and latency has been performed for evaluation purpose.

The rest of the paper is organized as follow: In next section, we introduce main works in literature related to energy saving at MAC layer level. Some OSI-based protocols and others based on Cross-layering approach are given in this section. In section 3, we give more details about the proposed protocol CL-MAC. In Section 4, we present a formal representation of CL-MAC protocol using time Petri nets modeling approach. Analyse and performance evaluation of CL-MAC protocol are presented in section 5. Finally, we conclude our work and discuss some future perspectives.

2. RELATED WORK

In this section, we present some MAC layer protocols developed some years ago enabling energy conservation in WSN. Firstly, compatible OSI protocols are presented and followed by two important cross-layer protocols: MAC-CROSS and XLM. Especially, MAC-CROSS protocol is considered as a basis of the development of our proposal.

2.1 Compatible OSI protocols

Many Studies in WSN showed that energy consumption during a communication is four greater than the energy consumed in processing and sensing operations together. This fact leads communication protocols designers to take a particular interest to WSN-MAC layer and to propose some original ideas to manage efficiently this layer. The medium access must take into account all sources of energy wastage considerations.

S-MAC or Sensor-MAC is a very popular protocol developed at California University [5, 6, 1] and is considered as a MAC protocol referenced by the WSN-scientific community. The principle of S-MAC protocol is based on a number of frame delays entering in a communication. Each frame is divided into two equal periods: wake-up and sleep periods. Synchronization of nodes takes place in a neighbourhood' node and exchange their calendars. A wake-up period enables for two neighbour nodes in a wake-up state to remain active during all the duration of the communication. In the sleep period, when a communication ends, communicating nodes enter in sleep mode by switching off their transceivers. This mechanism makes it possible to prolong network lifetime.

Several energy efficient protocols in the literature are based on Wake-up/Sleep mechanism: T-MAC [6], Z-MAC [2], X-MAC and B-MAC [7]. T-MAC and B-MAC are considered like S-MAC protocol with adaptive Wake-up/Sleep periods. In these protocols, as early as a node detects that a medium is occupied, enter immediately in sleep mode so as to preserve more energy and reduce a collision risks in his neighbourhood.

2.2 Cross-layer protocols

Other protocols based on OSI layer models try to reduce problems encountered in WSN. Network layer protocols tend to optimize paths between network' nodes and the Sink while application layer protocols try to obtain correct, accurate and compressed and/or aggregated information [8, 9] so as to reduce the amount of packets in the network. OSI-based protocols are not flexible, not optimal and consequently reduce network performances. To mitigate these drawbacks, a new MAC approach based on interaction or unification of two or more adjacent layers, called Cross-layer MAC optimization, is emerged [10, 4, 11]. Some protocols using a Cross-layer technique in medium access control layer can be found in literature such as MAC-CROSS [4] and XLM [11].

2.2.1 MAC-CROSS protocol

In MAC-CROSS protocol, only a few nodes concerned of the actual data transmission are asked to wake-up, while other nodes that are not included on a routing path and hence not involved in the actual transmission at all. In exchanging *RTS* and *CTS* packets, a field corresponding to a final destination address is added. The neighbourhood nodes belonging to the path extend their wake-up time while other nodes prolong their sleep time. Destination address is given by a routing table in the network layer. A similar mechanism to ARP protocol of IP network has been proposed if different addressing mechanisms are used in MAC and routing layers separately (see Figure 1).

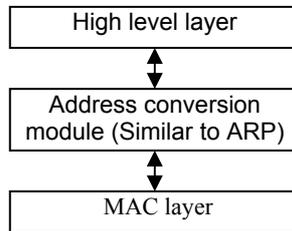


FIGURE 1: Address conversion scheme

2.2.2 XLM protocol

XLM (*Cross-Layer Module for Wireless Sensor Networks*) proceeds differently comparing to others traditional architecture based protocols for WSN. The communication in XLM is based on *initiative* concept considered as the core of XLM and implicitly incorporates the intrinsic functionalities required for successful communication in WSN. A node starts a transmission by transmitting to his neighbourhood an *RTS* packet to indicate that it has a packet to send. Upon receiving an *RTS* packet, each neighbourhood node decides to participate to communication by determining an *initiative I* defined as follows:

$$I = \begin{cases} 1, & \text{if } \begin{cases} \xi_{RTS} \geq \xi_{Th} \\ \lambda_{relay} \leq \lambda_{relay}^{Th} \\ \beta \leq \beta^{\max} \\ E_{rem} \geq E_{rem}^{\min} \end{cases} \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

ξ_{RTS} : Signal Noise Ratio (SNR) received of an *RTS* packet,

λ_{relay} : packets rate transmitted by relay and by node,

β : node's buffer occupation,

E_{rem} : node's residual energy.

Values in right side of inequalities give us respective thresholds and initiative *I* is initialised to 1 if all conditions illustrated in (1) are satisfied.

1. First condition ensures that reliable links are built for the communication.
2. Second and third conditions are employed by local control congestion. Second condition prevents congestion by limiting the transmitted traffic by a relay node, while the third condition ensures that no buffer overflow exists for this node.
3. Last condition ensures that the residual energy E_{rem} of a node do not exceed a minimal threshold E_{rem}^{\min} .

Cross-layering functionalities of XLM protocol are represented by the constraints defined in the initiative *I* of a node enabling it to carry out a local congestion control, hop by hop reliability and distributed operation.

The component of local congestion control of XLM ensures energy efficiency and a reliable communication. Results of performance evaluation revealed that XLM is better than one-layer protocols in terms of communication processing and implementation complexity considerations.

3. CL-MAC PROTOCOL PRESENTATION

CL-MAC (*Cross-Layer MAC*) can be added to MAC protocols class exploiting interactions between adjacent layers in order to minimize all energy wastage sources. The MAC layer enables access to the medium with an adaptive Wake-up/Sleep calendar. Nodes are locally and periodically synchronised like in Z-MAC protocol. The network is randomly deployed and nodes take a geometrical form (grid or linear) followed by a synchronisation phase.

The neighbourhood list established by each node contains information about neighbouring nodes (identifier, position, and calendar) and the routing table maintained by a routing agent in adjacent network layer. CL-MAC operates at MAC layer level by exploiting routing information using a similar mechanism adopted by MAC-CROSS protocol. For this reason, weak modifications are made in *RTS* and *CTS* message structures without violate the IEEE 802.11 standard. The modified message structures of *RTS* and *CTS* illustrated by figure 2 are proposed. The Sink node address is supposed to be known at the level of each node of the network in case of a mono-Sink WSN.

<i>RTS</i>	Frame control	Delay	Previous node address	Next node address	Sender address	CRC
<i>CTS</i>	Frame control	Delay	Previous node address	Next node address	Sender address	CRC

FIGURE 2: Structure of *RTS* and *CTS* messages (■ added fields)

However, it is important to note that *CTS* and *RTS* messages have identical structure but the difference is in their interpretation by the receiver node. Thus, if the receiver is:

- *next node*: the message acts as an *RTS* sent by the sender.
- *previous node*: the message acts as a *CTS* coming from a transmitter as a response to an *RTS*.
- *any other node*: the message controls the behaviour of the node and forces it to put into sleep mode.

In a wake-up period, a node continues to listen the medium for a short period. Two cases can be considered if the medium is not allocated:

Case 1. The node has data to transmit towards the sink, then take possession of the medium immediately and informs its neighbour nodes of this decision.

Case 2. Node does not have data to transmit. In this situation, it runs out its transceiver in order to avoid energy dissipation in idle listening and overhearing situations. In this way, the node takes more time in sleep mode than in S-MAC protocol ($\frac{1}{2}$ frame + communication time). If in this period the medium is allocated, this means that a communication is occurring or another node trying to obtain the medium control. In this case two situations can occur again:

- *Situation 1.* Another node tries to transmit packets. Then, Backoff algorithm [12] is used to resolve this contention problem and make possible only one node (elected node) to obtain the control access right. All other nodes are putting into sleep mode except the receiver of the packet which remains awake so to communicate with the elected node.
- *Situation 2.* The node has no data to send. Then, if it is concerned by the communication it remains in wake-up mode, otherwise it is putting into sleep mode until the next communication.

A receiver node is identified by the sender by referring to its routing table. This table contains all information related to the path between the sender and the Sink (distance, hops number of each alternative path, previous node identifier). In our case, the short path approach is implemented in network layer by applying Dijkstra- Moore algorithm.

Our protocol forces all nodes that are not selected for routing paths to put into sleep mode. In the network, the routing path appears as a lighted path from the source node to the Sink. This idea on which our protocol CL-MAC is based enables to eliminate all sources of energy wastage described in section 1.

The Figures 3 and 4 below illustrate the main advantage of CL-MAC protocol comparing to others MAC-protocols like S-MAC and MAC-CROSS.

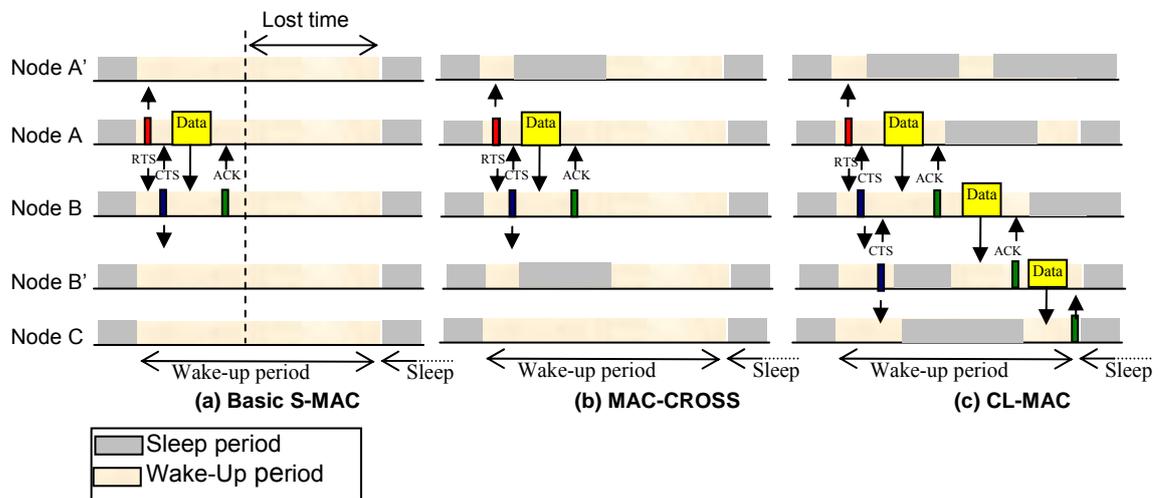


FIGURE 3: Main functionalities of S-MAC (a) MAC-CROSS (b) and CL-MAC (c)

As explained above, in S-MAC protocol, a frame is divided into two fixed periods: one for a listening who causes a useless loss of energy consumption and another for sleeping. In listening period, the control messages *CTS/RTS* are exchanged. The CSMA/CA mechanism is used for packet transmission in order to announce to a source node the next data transmission, for example, source node *A* in figure 3. It is clear that a node like *B'*, not concerned by the communication (between nodes *A* and *B*), must enter into sleep mode during all communication time or after, according to the listening period fixed by the protocol. Just after the end of communication, node *B* can start another communication without waiting the expiration of sleep period, e.g. waiting the next frame. This was adopted for designing adaptive S-MAC protocol. In each frame, MAC-CROSS protocol enables the exchange of *RTS/CTS/ACK* messages between three consecutive nodes using routing information obtained from the routing agent in network layer, while CL-MAC does the exchange of the same messages but only at the beginning of each frame, then the third node transmits however directly the received *CTS* to the next node in the path. This *CTS* message is interpreted in case of CL-MAC by a receiver node as a *RTS* of normally the next frame in MAC-CROSS protocol, for each routing node to another in the path until arriving to the Sink during the same frame. This allows all nodes belonging to routing path to remain active (or mobilized) until each node transmits successfully data packets to the next node. Just after transmitting these data packets, the node either enters into a sleep mode or prepares to begin a new transmission frame as a new source if it has data to send. Therefore, the frame duration in case of CL-MAC protocol is equal to time taken by a transmission from the source until the Sink.

4. MODELING CL-MAC PROTOCOL

In order to validate formally our protocol, it is important to model it in a convenient mathematical model according to CL-MAC specifications. The introduction of time Petri nets [13, 14] is motivated by their ability to model easily temporal constraints and the existence of a TiNA analyser tool for properties verification.

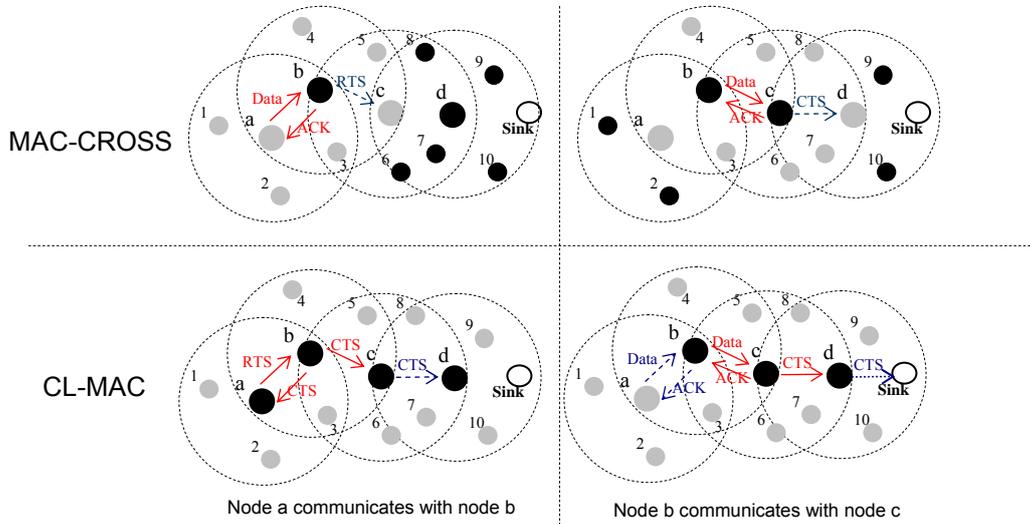


FIGURE4: Main advantages of CL-MAC protocol

Time Petri nets or TPN for short are a convenient model for real time systems and communication protocols [15]. TPN extend Petri nets by associating two values (min, max) of time (temporal interval) to each transition. The value min ($\min \geq 0$), is the minimal time that must elapse, starting from the time at which transition t is enabled until this transition can fire and max ($0 \leq \max \leq \infty$), denotes the maximum time during which transition t can be enabled without being fired. Times min and max, for transition t , are relative to the moment at which t is enabled. If transition t has been enabled at time α , then t cannot fire before $\alpha + \min$ and must fire before or at time $\alpha + \max$, unless it is disabled before its firing by the firing of another transition [15].

Figure 5 gives the time Petri net model of CL-MAC protocol. The values in the intervals associated to transitions refers to relative time to transmitting packets (RTS, CTS, DATA, ACK) according to IEEE 802.11 standard [16] and respected by our proposal. Hypotheses on the behaviour of CL-MAC are as follows:

- SIFS duration = 1 time unit
- Control messages RTS, CTS and ACK consume 3 time units
- Data requires 10 time units for its transmission
- Initially, only the places p1, p8, p14 and p17 are marked by one token.

The meanings of the transitions are given by Table 1.

5. CL-MAC PROTOCOL EVALUATION

To evaluate CL-MAC, we have performed some experimentations using TiNA tool for verification of its formal properties and using C++ software for studying its energy efficiency and latency.

5.1 Formal properties verification of CL-MAC

An enumerative method exists in order to exhaustively validate the behaviour of TPN models. This technique is related to the reachability analysis method for usual Petri nets and allows one to formally verify time dependent systems. The TiNA tool (Time Net Analyzer) version 8.0 available at (www.laas.fr/bernard/tina.html) uses the enumerative method and is applied to the verification of TPN model of CL-MAC protocol illustrated by the figure 5. The analyse reveals that CL-MAC protocol has good properties (live, bounded and reversible). The Figure 6 summarizes some results given by *Net Draw Control* of TiNA tool.

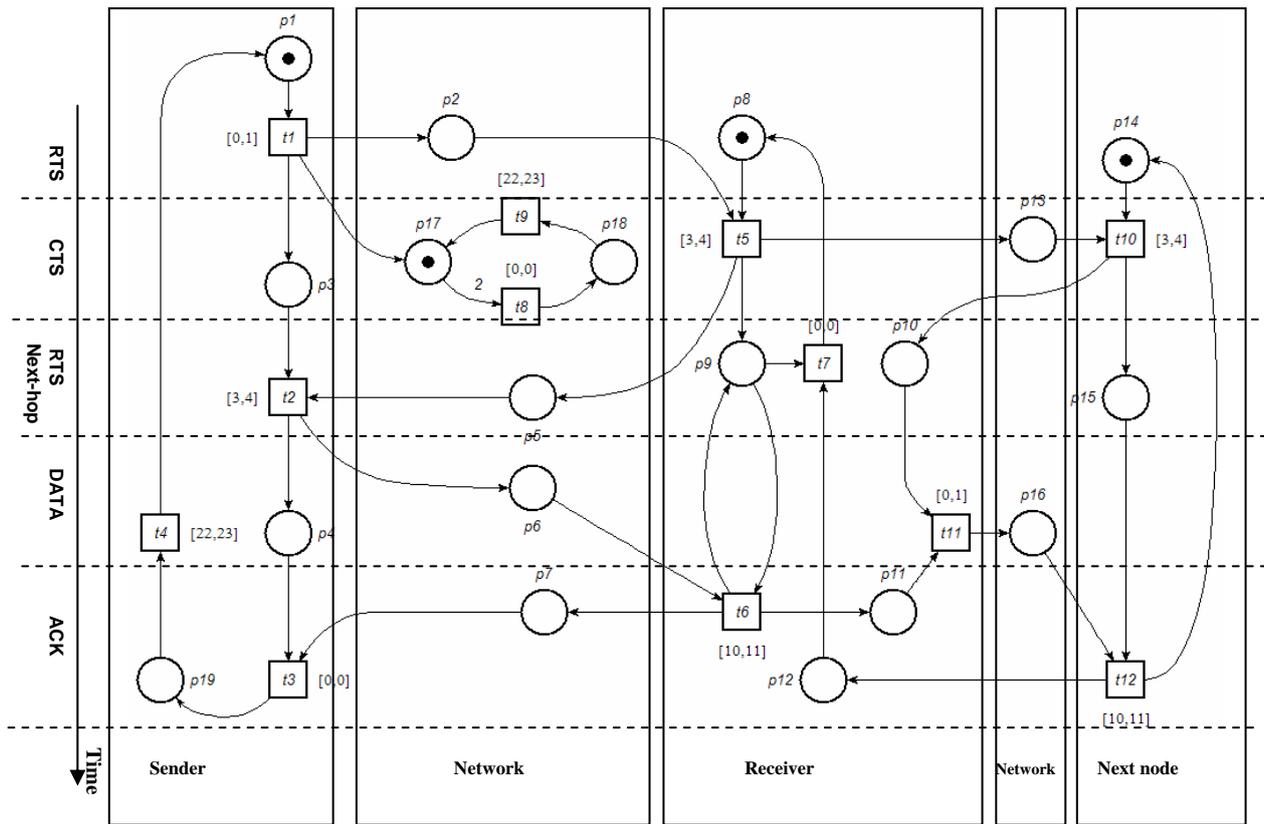


FIGURE 5: Time Petri net model for CL-MAC protocol

Transition	Meanings
t1	Transmission of <i>RTS</i> packet in unicast mode but the nature of the medium acts as broadcast
t2	Transmission of <i>DATA</i> packet in unicast mode
t3	Sender node enters into sleep mode
t4	Awakening of the sender to start a new frame
t5	Transmission of <i>CTS</i> packet in multicast mode by the receiver
t6	Transmission of <i>ACK</i> packet in unicast mode after receiving <i>DATA</i> packet
t7	Entering into sleep mode of the receiver after achieving his communication (frame)
t8	Entering into sleep mode of neighbour node not concerned by the actual communication
t9	Awakening of neighbour node not concerned by the actual communication after the last one ends
t10	Transmission of <i>CTS</i> packet by the next node
t11	Transmission of <i>DATA</i> packet by the receiver to the next node
t12	Entering into sleep mode of the next node

TABLE 1: Model transitions and their corresponding explanation

```

LIVENESS ANALYSIS .....
Possibly live
0 dead classe(s), 19 live classe(s)
0 dead transition(s), 12 live transition(s)
STRONG CONNECTED COMPONENTS:
0: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
SCC GRAPH:
0 -> t1/0, t8/0, t5/0, t10/0, t2/0, t6/0, t11/0, t3/0, t9/0, t12/0, t7/0, t4/0
0.000s
    
```

FIGURE 6: Analysis results given by TiNA tool

5.2 Performance evaluation of CL-MAC protocol

Performance evaluation of our protocol includes some metrics such as energy, delay and scalability. Because of the lack in practice of simulation tools dedicated for the Cross-layer protocols class, we have developed our simulation software using C++ Builder programming language. In this software, we have implemented at the same time our protocol CL-MAC and S-MAC. The kernel of the simulator is event oriented and based mainly on object oriented paradigm. In the future, we hope to adapt the NS2 (Network Simulator) software in order to support simulation of this class of protocols.

5.2.1 Simulation Environment

An example of deployed WSN network generated by our simulator and used for CL-MAC evaluation is illustrated by the Figure 7 and simulation parameters are summarized in Table 2. Latency (μ second) used in our experimentations is defined as an elapsed time between the time of message sending by a source node and the time of arrival of this message to the final destination (Sink). For the calculus of energy consumption, we have used a first order energy model introduced by [17].

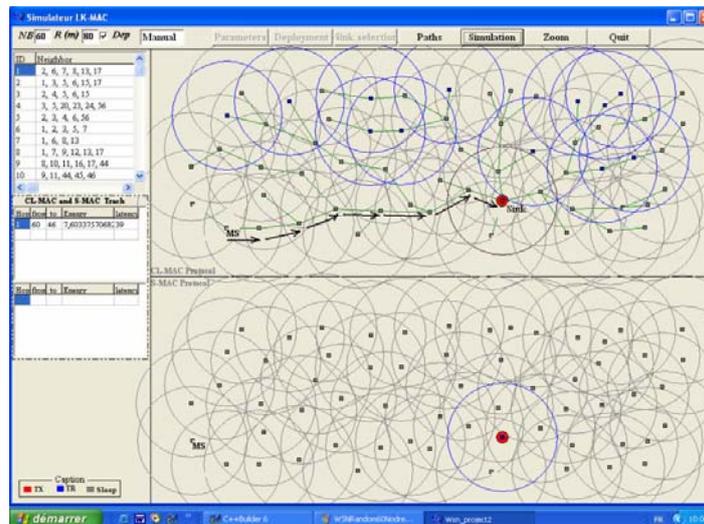


FIGURE 7: Example of WSN with 60 sensor nodes. The simulation interface is divided into two windows: the top window for CL-MAC and the bottom window for S-MAC

Parameter Type	Test Value
Number of sensor nodes (in the same deployment space)	Changes according to the evaluation example
Number of Sink	01
RTS/CTS Message size	118 bits
ACK message size	112 bits
Data message size	800 bits
Throughput	8 bits/ μ s

TABLE 2: Simulation parameters

5.2.2 Performances analysis

The aim of simulation is to analyze the effect of variation of some parameters such as number of data sources, density and hop number on CL-MAC and S-MAC behaviour in terms of dissipated total energy and latency in the network. Figure 8 shows that if the number of data sources increases in the network, the total energy consumed by S-MAC increases more quickly than that consumed by CL-MAC. But, more than 46 active data sources simultaneously make our protocol more consuming energy. This can be explained by the fact that each time a data source is added to the network, more nodes will be mobilized to remain in active state (thus their duty cycle increases) in order to participate in routing paths as intermediate nodes. Thus, there will be fewer nodes in Sleep state (but the effectiveness of our protocol is based on its ability to put into Sleep mode any node not concerned by the routing operation). On the other hand, S-MAC has the capability to put into Sleep mode each sensor node for a half cycle independently of source nodes number. Therefore, no changes will be made in node duty cycle.

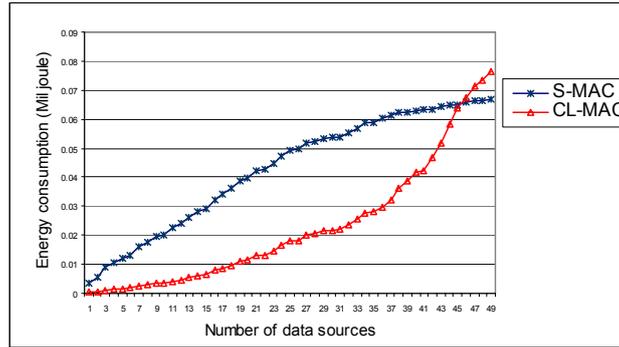


FIGURE 8: Energy consumption according to the number of data sources with 60 deployed nodes

In the experimentation producing the results of Figure 9, we have varied the density of the network in a fixed deployment space and we have chosen a data source located at 10 hops far from the Sink. The data source sends data packets to the Sink on the routing path. As the number of neighbours of each node belonging to the path changes randomly according to the density, we have chosen the average number of these neighbours in numerical evaluation of total energy consumed by the network. In Figure 9, CL-MAC always maintains its best performance as a protocol with a minimum energy consumption compared to S-MAC. This last one consumes more energy if the network density increases.

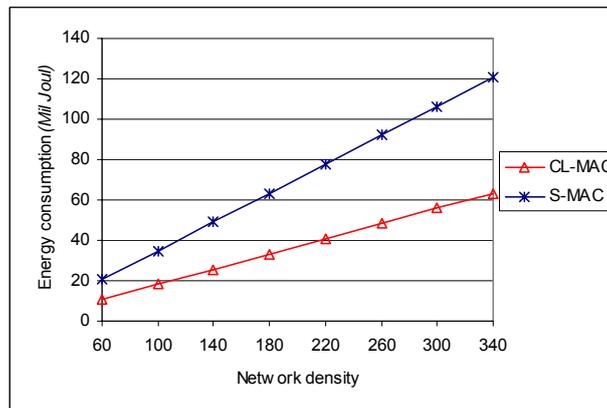


FIGURE 9: Energy consumption in case of varying density

Figure 10 shows well that latency quickly increases in the case of CL-MAC when a new source node is added to the number of nodes already in communication. On the other hand we note, for a given number of sources lower than 4, CL-MAC records latency better than S-MAC. This can be explained by the nature of the progressively generated paths. Indeed, in the presence of not disjoint paths (paths having at least one intermediate node in common), CL-MAC protocol leaves in active mode only one node included in a routing path and puts the remainder of neighbour nodes into Sleep mode. However, if this active node is included in several routing paths, then the only path, which will be operational, is that will have possession of this active node. The other paths will be delayed until the current routing step at this active node is finished. More precisely, the neighbour nodes of the active node, that have been included in other routing paths, will be found in Sleep mode at the moment when the nodes, which precede them in their corresponding paths, want to transmit a message to them. It is consequently clear that in the case of disjoint paths, CL-MAC maintains its performances although the insurance of a presence of disjoint paths in the network from many sources is in practice a very challenging issue.

The results of Figure 11 are obtained by carrying out 5 transmissions of data packets for each hop where the resulting average value is considered. We note in this figure that each time a sensor node moves away from the Sink, CL-MAC protocol records a better latency compared to S-MAC. Indeed, while moving away from the Sink, the routing path will contain more sensor nodes to participate in routing operation. Therefore, according to the strategy adopted in CL-MAC, there will be more potential sensor nodes to put into Sleep mode. This explains the gain in term of latency in presence obviously of only one routing path at a given time.

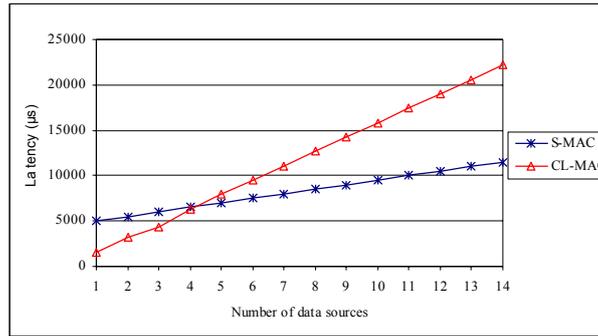


FIGURE 10: Latency vs number of data sources with 150 deployed nodes

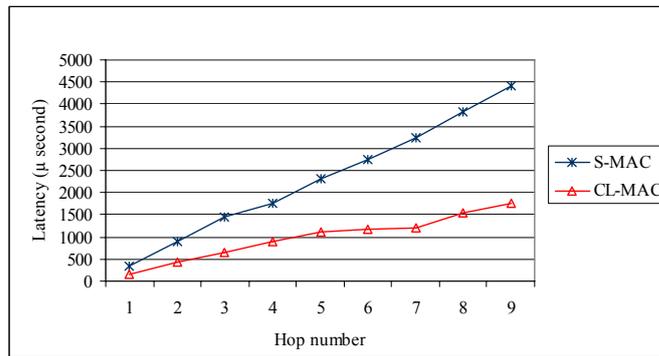


FIGURE 11: Latency vs hop number with 60 deployed nodes

5.2.3 Idle listening

This source of energy wastage cannot exist when CL-MAC protocol is used. Figure 12 below shows different states of a node when CL-MAC protocol is applied: Sending (TX), Receiving (RX), Sleeping and Accessing to medium.

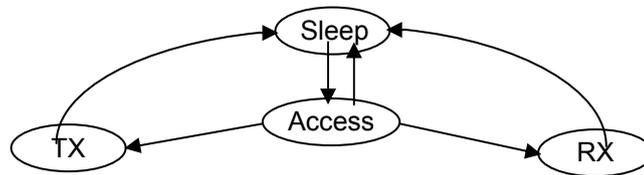


FIGURE 12: State transition diagram of sensor node

5.2.4 Scalability

Protocol CL-MAC enables to a network composed of a number of nodes, which can reach 1000 sensors nodes, to normally operate in different experiments that we have performed, especially in energy saving study, without degradation of its performances.

6. CONCLUSION AND FUTURE WORK

In this paper, we have proposed a novel energy efficient MAC protocol using a Cross layer optimization approach, named CL-MAC. This protocol has been modeled using time Petri nets and validated using TiNA tool. The simulation experiments undertaken on several network topologies of wireless sensor networks showed the effectiveness of CL-MAC protocol in terms of energy consumption and latency. Other interesting characteristics of this protocol, that we can found in the majority of MAC protocols to energy saving, allow to mitigate some problems such as collision and idle listening which has been proved to be potential sources of energy wastage; and in addition support the scalability. The comparison carried out with S-MAC protocol is justified by the fact that S-MAC is regarded as a reference chosen by the community of researchers for studying MAC layer behaviour for WSN.

Various experiments that we have performed show that CL-MAC is largely better than S-MAC in term of energy consumption and latency in the following cases:

- when the data source is far from the Sink,
- when, at a given time, the number of data sources is low with no limit for network size.

These two cases characterize, in our opinion, the class of wireless video sensor network applications for surveillance of intruders, where a deployment field is in general far from the collection and decision-making station. Because of the reactive mode of this type of applications when an intruder detection event occurs, a video sensor node in general, or sometimes a restricted number of nodes carrying out the surveillance tasks in cooperation fashion, covering the corresponding area will become a video data source to transmit data stream without significant delay towards the collection center. The energy and the latency can be considered in this case as metrics of quality of service which can be guaranteed at the MAC level by employing Cross-layer approach. For this reason, these results encourage us to investigate the applicability aspect of CL-MAC protocol for this real applications category.

REFERENCES

- [1] Zhi-Wen, O., Shruthi, B. K. and Sang, S. K. (2005) Medium Access Control for Wireless Sensor Networks. *CS258 - Advanced Communication Networks*, San Jose State University.
- [2] Injong, R., Ajit, W., Mahesh, A. and Jeongki, M. (2005) Z- MAC: An Hybrid MAC for Wireless Sensor Networks. Dept. of Computer Science, North Carolina State University, *SenSys'05*, November 2–4.
- [3] Sungrae, C., Kanuri, K., Jin-Woong, C., Jang-Yeon, L. and June, S.-D. (2005) Dynamic Energy Efficient TDMA-based MAC Protocol for Wireless Sensor Networks. *Autonomic and Autonomous Systems and International Conference on Networking and Services, ICAS-ICNS*, Vol , Issue , 23-28, P.48 - 48
- [4] Suh., C., Young-Bae, K. and Dong-Min, S. (2006) An Energy Efficient Cross-Layer MAC Protocol for Wireless Sensor Networks. Graduate School of Information and Communication, Ajou University, Republic of Korea. *APWeb 2006, LNCS 3842*, pp. 410–419.
- [5] Wei, Y., Heidemann, J., Estrin, D. (2002) An energy-efficient MAC protocol for wireless sensor networks. *Proceedings of INFOCOM 2002, Twenty-First Annual Conference of the IEEE Computer and Communications Societies*, IEEE
- [6] Koen, L. and Gertjan, H. (2004) Energy-Efficient Medium Access Control. *The embedded Systems Handbook*. Delft University of Technology, Faculty of Electrical Engineering, Mathematics and Computer Science Mekelweg 4, 2628CD Delft, The Netherlands.
- [7] Buettner M., Gary Y., Eric, A. and Richard, H. (2006) X-MAC: A Short Preamble MAC Protocol For Duty-Cycled Wireless Sensor Networks, *Department of Computer Science, University of Colorado*, National Science Foundation.
- [8] Holger, K., Marc, L. and Tim, N. (2003) A Data Aggregation Framework for Wireless Sensor Networks. *European research project EYES*, Berlin.
- [9] Cheng, H., Qin, L. and Xiaohua J. (2006) Heuristic Algorithms for Real-time Data Aggregation in Wireless Sensor Networks. School of Computing, Wuhan University, China. *IWCMC'06*, July 3–6.
- [10] Akyildiz, I. F. and Ismail, H. K. (2004) Wireless sensor and actor networks: research challenges, Georgia Institute of Technology, www.sciencedirect.com.
- [11] Akyildiz, I. F., Vuran, M. C. and Akan O. B. (2006) A Cross layer protocol for wireless sensor networks. *Proc. Conference on Information Sciences and Systems (CISS'06)*, Princeton, NJ.
- [12] Ignatius, M. (2006) Energy-efficient Wireless Sensor Network MAC Protocol. *PhD Thesis in Electrical Engineering*, March 31, Faculty of Virginia Polytechnic Institute and State University.
- [13] Merlin, P. M. (1974) A Study of the Recoverability Systems of Computing. Irvine Univ. California, *PhD thesis*.
- [14] Berthomieu, B. and Menasche, M. (1983) An Enumerative Approach for Analyzing Time Petri Nets. *IFIP Congress Series*, Vol. 9, pp. 41-46 North Holland.
- [15] Berthomieu, B. and Diaz, M. (1991) Modeling and Verification of Time Dependent Systems Using Time Petri Nets. *IEEE Trans. On Soft. Eng.*, Vol. 17, n° 3, pp. 259-273.
- [16] IEEE Std. 802.11- (1999), Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, ISO/IEC 8802-11:1999(E), IEEE Std. 802.11.
- [17] Wendi, R. H., Chandrakasan, A., Balakrishnan, H. (2000) Energy-Efficient Communication Protocol for Wireless Microsensor Networks. *The Hawaii International Conference On System Sciences*, January 4-7, MAUI, HAWAII.