

Article

Energy Efficient Routing and Node Activity Scheduling in the OCARI Wireless Sensor Network

Saucene Mahfoudh, Pascale Minet * and Ichrak Amdouni

Institut National de Recherche en Informatique et en Automatique (INRIA), Rocquencourt, 78153 Le Chesnay Cedex, France; E-Mails: saucene.mahfoudh@inria.fr (S.M.); ichrak.amdouni@inria.fr (I.A.)

* Author to whom correspondence should be addressed; E-Mail: pascale.minet@inria.fr;
Tel.: +33139635233; Fax: +33139635566.

Received: 22 July 2010 / Accepted: 11 August 2010 / Published: 17 August 2010

Abstract: Sensor nodes are characterized by a small size, a low cost, an advanced communication technology, but also a limited amount of energy. Energy efficient strategies are required in such networks to maximize network lifetime. In this paper, we focus on a solution integrating energy efficient routing and node activity scheduling. The energy efficient routing we propose, called EOLSR, selects the route and minimizes the energy consumed by an end-to-end transmission, while avoiding nodes with low residual energy. Simulation results show that EOLSR outperforms the solution selecting the route of minimum energy as well as the solution based on node residual energy. Cross-layering allows EOLSR to use information from the application layer or the MAC layer to reduce its overhead and increase network lifetime. Node activity scheduling is based on the following observation: the sleep state is the least power consuming state. So, to schedule node active and sleeping periods, we propose SERENA that colors all network nodes using a small number of colors, such that two nodes with the same color can transmit without interfering. The node color is mapped into a time slot during which the node can transmit. Consequently, each node is awake during its slot and the slots of its one-hop neighbors, and sleeps in the remaining time. We evaluate SERENA benefits obtained in terms of bandwidth, delay and energy. We also show how cross-layering with the application layer can improve the end-to-end delays for data gathering applications.

Keywords: wireless sensor networks; energy efficiency; energy efficient routing; node activity scheduling; cross-layering; graph coloring

1. Context and Motivations

Wireless sensor networks consist of a large number of wireless sensor nodes that organize themselves into multi-hop radio networks. Sensor nodes are able to monitor a wide variety of physical parameters such as temperature, humidity, light, radiation, noise, *etc.* These sensor nodes have a low cost, small size and wireless data transfer capability. Hence, sensor networks are expected to find widespread use in applications such as environment protection, health monitoring, diagnosis help in maintenance applications, traffic monitoring on freeways or urban street intersections, seismic data-gathering. The sensor nodes are typically equipped by power-constrained batteries, which are often difficult, expensive and even impossible to be replaced once the nodes are deployed. Therefore, energy awareness becomes the key research challenge for sensor network protocols.

In fact, several researchers have addressed energy conservation aiming at maximizing the network lifetime. The energy consumed by a node depends on its state. We distinguish four states: (1) *transmit*, (2) *receive*, (3) *idle* when the node keeps listening the medium even when no messages are being transmitted and finally (4) *sleep*, where the radio module is switched off: no communication is possible. Table 1 reports the power used in each state for both IEEE 802.11 and IEEE 802.15.4 nodes.

Table 1. Power value in each radio state.

State	Power (Watts)	
	IEEE 802.11	IEEE 802.15.4
Transmit	1.3	0.1404
Receive	0.9	0.1404
Idle	0.74	0.0018
Sleep	0.047	0.000018

We can refine the receive state as follows:

- *overhearing*: when a sender transmits a packet, all its neighbors will receive this packet even if it is intended for only one of them. Thus, in the overhearing state, any one hop neighbor of the sender that is not the destination will dissipate energy;
- *interference*: each node located between transmitter range and interference range receives the packet but can not decode it;

More generally, energy efficient techniques tend to minimize the energy wasted in useless states from the application point of view such as idle listening, overhearing, interference, collision. Consequently, these techniques increase the network lifetime and improve energy efficiency. We can classify these techniques into four classes:

- *energy-aware routing protocols* select routes that minimize the energy consumed by an end-to-end transmission or visit nodes with high residual energy;
- *node activity scheduling algorithms*, whose objective is to turn off the sensor radio when it does neither transmit nor receive data;

- mechanisms to reduce the amount of data transferred like data aggregation, because the energy consumed depends on the data size;
- topology control methods tuning node transmission power.

We now focus on energy consumption in wireless *ad hoc* and sensor networks. More precisely, we quantify the energy dissipated in transmitting, receiving, overhearing, idle listening and interference during network lifetime. The network lifetime is defined as the time of the first sensor node failure. The simulation parameters are given in Table 2.

Table 2. Simulation parameters.

	Simulation parameter	Value
Configuration	Number of nodes	50–200
	Density	10
	Bandwidth	2 Mbps
	Transmission range	250 m
Energy	Initial energy	50 Joules
	Transmit	1.3 Watt
	Receive	0.9 Watt
	Idle	0.74 Watt
	Sleep	0.047 Watt
Traffic	Number of flows	30
	Throughput	16 Kbps
	Packet size	512 bytes
Routing	Hello period	2 s
	TC period	5 s

Figure 1. Distribution of node energy consumption without the sleeping state.

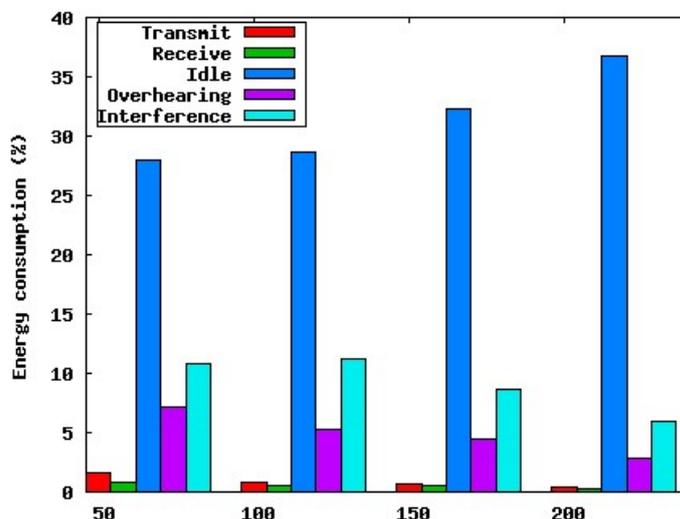


Figure 1 depicts the distribution of node energy consumption in each radio state, assuming that nodes are not allowed to sleep: they are in the idle state when they have nothing to transmit and nothing to receive. The OLSR routing protocol is used. A network of 100 nodes with a density of 10 is considered. The density denotes the average number of neighbor per node + 1. Each result depicted in Figure 1 corresponds to the average of 10 simulations. We notice that the highest part of node energy is wasted in useless states (states that do not correspond to user data reception or transmission). That is why, in this paper, we will focus on energy efficient routing and node activity scheduling to reduce the energy lost in the interference and idle states.

In this paper, we focus on an energy efficient solution for wireless sensor networks integrating an energy efficient routing and a node activity scheduling. For energy efficient routing, we have proposed EOLSR in [1,2]. The idea consists in reducing the energy consumed in an end-to-end transmission while avoiding nodes with low residual energy. New simulation results compared to [3] show that the increase in network lifetime is accompanied by an increase in the amount of data delivered to the user. In this paper, we show how the use of cross-layering between EOLSR and the application layer considerably reduces the routing overhead. For node activity scheduling, we use SchEduling RoutEr Nodes Activity (SERENA) based on node coloring. SERENA assigns a color to each node while using a small number of colors and ensuring that two nodes with the same color can transmit without interfering. SERENA has been presented as a two-hop coloring algorithm in [4–6]. In this paper, SERENA is generalized to support different application requirements (e.g., broadcast, data gathering applications, immediate acknowledgement). We evaluate the cost to support such application requirements. This cost is expressed in terms of additional colors required as well as additional messages needed in the coloring process. Another new contribution of this paper is the comparison between SERENA and TDMA-ASAP, evaluating the end-to-end delays and the potential color conflicts. Concerning the integration of EOLSR and SERENA, we evaluate in this paper, the amount of data delivered using EOLSR and/or SERENA (three-hop coloring). We also show that node energy is used more efficiently. For instance, no energy is lost in interferences, the energy lost in idle state is considerably limited. This integration is used in the OCARI project focusing on wireless sensor networks in industrial environment. We provide a refined version of the OCARI stack, initially presented in [7] and describe the strong interaction of EOLSR and SERENA with the MAC layer.

The remaining of this paper is organized as follows. In Section 2, we present a state of the art dealing with energy efficient techniques in wireless sensor networks, energy efficient routing, node activity scheduling, and cross layering. In Section 3, we give the principles of EOLSR and report its performance evaluation results. We also explain how cross-layering with both the application layer and the MAC layer can improve benefits obtained from EOLSR. In Section 4, we describe SERENA and evaluate its performance compared to existing solutions like TDMA and TDMA-ASAP. We then quantify the benefits brought by cross-layering to improve SERENA. In Section 5, we first present the benefits of the integration of SERENA and EOLSR regarding the amount of data delivered for the end user and the network lifetime. Second, we describe how SERENA and EOLSR are used in an industrial environment, that is the OCARI project. Finally, we conclude in Section 6.

2. Related Work

In this section, we give a brief state of the art concerning energy efficient routing in Section 2.1, node activity scheduling in Section 2.2 and cross-layering optimization in Section 2.3.

2.1. Energy Efficient Routing

Routing in wireless sensor networks is very challenging due to their specific characteristics that distinguish them from wireless *ad hoc* networks. Routing protocols differ in the way they take into account limited capacities of sensor nodes as well as the application and architecture requirements. Our purpose is not to give an exhaustive list of existing protocols but rather to provide a classification illustrated by some examples [3]. A simple way to save energy is to reduce the overhead needed to transfer the user traffic. For that purpose, we distinguish:

2.1.1 Data Centric Protocols

Data-centric routing protocols decrease the amount of transferred data by querying sensor nodes based on their data attributes or interests. This assumes a data query driven model and a routing based on data attributes. For instance, in SPIN [11], sensor nodes inform each other about the data they have and the data they still need to obtain. Thus, interested nodes can query these data. However, this protocol does not ensure the end-to-end delivery of data if intermediate nodes are not interested in that data. In directed diffusion [12], the sink broadcasts an interest message to nodes, interested nodes reply with a gradient message. Both interest and gradient messages allow the sink to establish paths to sensor nodes.

2.1.2. Hierarchical Routing Protocols

These protocols consider the clustering in order to improve scalability and reduce the traffic. LEACH [13] is one of the first hierarchical routing protocols for sensor networks. It chooses the cluster head based on its signal strength and utilizes randomized rotation of local cluster heads to evenly distribute the energy load among the sensors in the network. Only cluster heads, whose optimal number is estimated to be 5% of the total number nodes, route data to the sink, which saves sensor nodes energy. However, LEACH assumes that any node is 1-hop away from the sink. TEEN [14] is another example of hierarchical routing protocols. It is also data-centric. It builds clusters of different levels until reaching the sink node. The cluster head broadcasts two thresholds to the nodes: only a value of the sensed attribute higher than the hard threshold can force the sensor node to transmit, and if this sensor was already transmitting, only changes higher than the soft threshold should be transmitted. This mechanism considerably reduces the number of transmissions. However, TEEN is not adapted to periodic queries. An extension called APTEEN [15] has been proposed.

2.1.3. Geographic Routing Protocols

They are based on the fact that if the destination of a query is known (sensors locations are assumed to be known, e.g., GPS), the query can be broadcast only to the appropriate geographical region without flooding. Hence, the amount of broadcast data is reduced. In GEAR [16] for instance, a node forwards a

packet to its neighbor that is the closest one to the destination. If all neighbors are at the same distance, some neighbor is randomly selected. GAF [17] is another example of energy aware geographical routing protocol. Indeed, each node uses the knowledge of its location provided by its GPS to determine its region in a virtual grid. The size of each cell in the grid is set such that any two nodes belonging to contiguous cells can communicate directly. Nodes located in the same cell are considered equivalent from the routing point of view. Furthermore, GAF allows nodes to sleep as long as the existence of an active node per cell in the virtual grid is achieved.

There are different ways to compute energy efficient routes. Routing protocols differ by:

2.1.4. Energy Criteria Taken into Account for Route Selection

These criteria are: (1) the minimum energy path like [18], (2) the path avoiding nodes with low residual energy like REAR [19], and (3) the path of minimum energy while avoiding nodes with low residual energy, like EOLSR (see Section 3).

Furthermore, a routing protocol can maintain a single or multiple path to reach a destination.

2.1.5. Multipath Routing Protocols

Multipath routing protocols [20,21], avoid the depletion of nodes energy along the path having the lowest energy by sharing the load on several paths. As energy is taken into account, the path minimizing the energy consumed by an end-to-end transmission is usually selected as the primary path. Another path, called the secondary one, is used less frequently, with a probability inversely proportional to its energy cost.

2.2. Node Activity Scheduling

Since the sleeping state is the least energy consuming state, keeping nodes in the sleeping state is a good way to save energy. However, this should be accompanied by node activity scheduling to prevent network partition and message loss when some nodes are sleeping. We can distinguish two types of solutions according to their dependency to the medium access.

2.2.1. Solutions Independent of the Medium Access

In this type of solution, nodes are assumed to be redundant to achieve the functionalities required by the application. The goal of these protocols is then to build active sets, sequentially activated. Only nodes belonging to the active set are active, all the other nodes are sleeping. When these sets are disjoint, network lifetime is equal to the sum of the active sets lifetimes. In [22], the authors show that maximizing the number of disjoint active sets is NP-complete. The authors of [22] and [23] propose two centralized solutions assuming synchronized nodes and the same energy consumption for all nodes in the same active set. They show in [23] that considering non disjoint sets improve the network lifetime. Distributed solutions also exist as in [24], where each node only needs to know information about its one-hop and two-hop neighborhood. A node decides to stay active or to sleep depending on its residual energy and the decision of its neighbors. More precisely, a node whose function is ensured by active nodes can sleep

if and only if (1) the active nodes are connected, (2) all its neighbors have at least one active neighbor and (3) all active nodes have a residual energy higher than its own.

2.2.2. Solutions Dependent on the Medium Access

With this second type of solutions, any node is allowed to sleep whenever it is neither transmitting nor receiving. These solutions can be organized in three classes depending on the medium access.

- *CSMA/CA*: This medium access provides spatial reuse of the bandwidth, is highly adaptive and has a low overhead in case of light load. All the solutions relying on CSMA/CA are based on the RTS/CTS exchanges preceding unicast transmissions. Any two nodes exchanging RTS/CTS packets need to keep active to start the actual data transmission, whereas their neighbors can enter the sleeping mode to avoid overhearing and idle listening. S-MAC [25] is a famous example. Many other variations of S-MAC have arisen such as T-MAC [26] with an adaptive duration of the active period, D-MAC [27] that reduces network latency, O-MAC [28] that improves the throughput. We can notice that RTS/CTS packets increase the overhead and reduce protocol efficiency. Hence, they are not adequate in case of short messages, the usual case in wireless sensor networks.
- *TDMA*: Since the transmissions are scheduled in slots, TDMA ensures that no collision will occur and hence saves energy. It provides a deterministic guarantee for the transmission delays. In order to save energy, bandwidth and delay, the active period during which each node can transmit must be kept as small as possible. Hence, several nodes will transmit in the same slot. To be valid, the schedule must ensure that all nodes allowed to transmit in the same slot do so without interfering. We can distinguish two types of slot assignment:
 - *slots are assigned per node*. In such a case, the transmitting node can use its slots as it wants: it can broadcast, send a unicast transmission to one of its neighbors then to another one. The advantage is that it can optimize the utilization of its slot. The drawback is that any neighbor of the transmitter must be awake during this slot, because it can be the destination of a message. Such solutions are also called broadcast scheduling as in [30] for instance. Krumke *et al.* have defined the problem called channel assignment in radio networks and have established complexity results for different types of network configurations [29]. They propose a 2-approximation algorithm for the minimum two-hop coloring on bounded degree planar graphs. With two-hop coloring, two nodes that are one-hop or two-hop neighbors must not have the same color. TDMA-ASAP [36] is designed for data gathering applications. Based on node coloring, it aims at providing spatial reuse, saving energy and decreasing the end-to-end delays. Moreover, this protocol considers slot stealing to adapt to various traffic conditions. However, it does not support the immediate acknowledgement of unicast transmissions between a parent and its child.
 - *slots are assigned per link*. In this case, only the transmitter and the receiver, the two extremities of the link are awake, all other nodes are sleeping. However, if the traffic on this link is light, the slot is not used at 100%. Broadcast transmissions are expensive: they require to copy the same information n times where n is the number of neighbors. Such solutions are also called link scheduling in [32] and [33]. A genetic solution is proposed

in [31]. Two examples of deterministic slot assignment per link are given by TRAMA for general communications, FLAMA for tree-based communications like in data gathering applications. TRAMA [34], consists of 1) a neighborhood discovery protocol, 2) a schedule exchange protocol and 3) an adaptive election algorithm that selects the transmitter and receiver(s) for each time slot. The node having the highest priority among its one-hop and two-hop neighbors wins the right to transmit in the considered slot. Each node declares in advance the list of its slots and for each slot its receiver(s). TRAMA is adaptive but also complex. To mitigate this complexity, a solution named FLAMA [35] is introduced for data gathering applications. FLAMA is simplified both in terms of message exchange and processing complexity.

- *Hybrid*: Z-MAC [37] is based on DRAND [38] which assigns slots to nodes in such a way that any node has a slot different from those assigned to its one-hop and two-hop neighbors. The goal of Z-MAC is to optimize the bandwidth utilization the MAC protocol, selecting CSMA/CA under low contention and TDMA under heavy contention. We can notice that Z-MAC does not allow an immediate acknowledgement of unicast messages, while this acknowledgement is important in wireless communication to confirm the correct reception of the packet. From the energy point of view, Z-MAC reduces the activity period in the polling cycle enforced by the application. It does not allow nodes to sleep during the activity period, unlike SERENA (see Section 4) that aims at maximizing network lifetime by scheduling node activity. The advantage of Z-MAC is that it does not depend on the number of network nodes but on the cost of an asymptotic convergence.

More generally, slot assignment can be based on a coloring algorithm, as in Z-MAC and TDMA-ASAP. In graph theory, graph coloring is a special case of graph labeling; it is an assignment of labels, traditionally called colors, to elements of a graph subject to certain constraints. Before presenting the two categories of graph coloring (edge or vertex coloring), we define the criteria used to evaluate a distributed coloring algorithm, [45,46]:

- the number of colors needed to color a graph G : closer this number to the chromatic number of G , more efficient the algorithm.
- its time complexity, expressed in the case of a distributed algorithm, by the maximum number of rounds needed to color each node and the total number of messages exchanged to color G .

The concept of round, used to evaluate the time complexity of a graph coloring algorithm, can be defined as follows. In a round, any node can:

- send a message to all its one-hop neighbors,
- receive the messages sent by them,
- perform some local computation based on the information contained in the received messages.

The two classes of graph coloring are vertex coloring and edge coloring.

Vertex coloring has received a lot of attention from researchers. One-hop vertex coloring consists in coloring each vertex of the graph such that two adjacent vertices have not the same color and the number of colors used is minimum. This problem has been shown NP-complete in [39] for the general

case, whereas graphs with maximum vertex degree less than four, and bipartite graphs can be colored in polynomial time. The first researches on one-hop graph coloring proposed centralized algorithms (see [40,41]). In [40], Dsaturn, a greedy algorithm (*i.e.*, no color backtracking), colors first the vertex with the highest number of already colored neighbor vertices. This algorithm exhibits very good performances, even if it is not optimal. It is then followed by Largest First, where the node with the highest degree is colored first. Distributed one-hop graph coloring algorithms also exist. Some of them resort to randomization to select the color as for instance [42–44]. The color selected by a node can be used only if it does not conflict with the colors chosen by its neighbors. Let n be the number of vertices and Δ the largest vertex degree. For one-hop graph coloring, the algorithm proposed in [45] runs in $O(\log n)$ rounds, and uses a number of colors close to Δ , whereas Distributed Largest First runs in $O(\Delta^2 \log n)$ rounds [46].

Edge coloring is an assignment of colors to the edges of the graph such that edges incident on the same vertex receive different colors. Note that an edge coloring of a graph is just a vertex coloring of its link graph. Many solutions focus on this type of graph coloring to apply it to the wireless networks. These solutions are named link scheduling algorithm. The idea is to assign to each link between two neighbors a color (which corresponds to a time slot) in which these two neighbors can communicate. The problem of assigning colors to edges for a general graph using the minimum number of colors is NP-complete [47]. Gandham *et al.*, [32] proposed a distributed edge coloring algorithm that needs at most $\Delta + 1$ colors, where Δ is the maximum degree of the graph. Each color is mapped to a time slot and a direction of the transmission is determined to avoid the problem of the hidden and exposed terminal. Other solutions [48,49] focus on the two-hop edge coloring wherein two edges are assigned different colors if they are adjacent or if they are connected by an edge.

2.3. Cross Layering Optimization

Cross layer design breaks away from traditional network design, where each layer of the protocol stack operates independently and exchanges information with adjacent layers only. In the cross layer approach, information is exchanged between non-adjacent layers of the protocol stack in order to optimize decisions and better adapt to the environment. Examples of cross-layering architectures are described in [50–54].

In [55], using a cross layer approach, authors study the tradeoff between network lifetime maximization and rate allocation problem by formulating these two problems as a constrained maximization problem. Using Lagrange dual decomposition, the original problem is vertically decomposed into three subproblems: (1) a rate control problem at the transport layer, (2) a contention resolution problem at the MAC Layer, and (3) a cross-layer energy conservation problem. Two fully distributed algorithms are derived for the first two subproblems. For the third subproblem, they first directly derive a partially distributed algorithm, and then propose a fully distributed approximation algorithm using network utility maximization framework. Nama *et al.* [56,57] characterized the tradeoff between maximizing the application performance and lifetime by considering a cross-layer design problem in a wireless sensor network with orthogonal link transmissions, which jointly maximized the network utility and lifetime. The idea is to compute an optimal set of source rates, network flows, and radio resources at the transport, network, and radio resource layers respectively, while jointly maximizing

the network utility and lifetime. They show that the cross-layer optimization problem decomposes both horizontally (across nodes) and vertically (across different layers in the protocol stack) into simpler subproblems allowing a fully distributed solution. In [58], authors investigate the problem of the lifetime maximization in a wireless sensor network under the constraint of the end-to-end transmission success probability. A cross-layering strategy that considers physical layer (*i.e.*, power control), MAC layer (*i.e.*, retransmission control) and network layer (*i.e.*, routing protocol) jointly is adopted. They first present a near-optimal retry limit allocation algorithm for a given routing path. This allocation determines per-hop success probability for each link along the path in order to minimize the total energy consumption while guaranteeing the reliability constraint. Then, an optimal routing and power control algorithm that maximizes the network lifetime is developed. Simulation results reveal that a trade-off relation exists between the network lifetime maximization and the reliability constraint, and that the network lifetime can be increased significantly by employing the proposed algorithms.

In this paper, we show in Section 3.3 how to optimize the EOLSR routing protocol, taking advantage of information provided by (i) the MAC layer (e.g. the received power to use only links of good quality), and the application layer (e.g., existence of a sink node). Similarly, we optimize SERENA in Section 4.4 taking advantage of cross-layering to reduce the end-to-end delays in data gathering applications.

3. Energy Efficient Routing

3.1. Principles

EOLSR, is the energy efficient extension of OLSR. Like OLSR, it consists of two main functionalities: neighborhood discovery achieved by exchanging *Hello* messages between one-hop nodes and topology dissemination achieved by broadcasting *TC* messages advertising preferred links throughout the network. EOLSR maximizes the network lifetime by:

- Minimizing the energy consumed by a packet transmission from its source to its destination. The transmission and reception is a source of energy consumption, and optimizing their number could optimize the energy consumption.
- Balancing load between nodes and avoiding nodes with a low residual energy. Using the same nodes to route messages exhausts the batteries of these nodes. As a consequence, they will fail more quickly than others. This could lead to network partitioning or some application functionalities are no longer assured (e.g. a zone is no more monitored).
- Reducing the overhead.

EOLSR uses an energy consumption model to take its routing decision. As MPRs (MultiPoint Relays) are the intermediate nodes on routes, the selection of MPRs must take energy into account as well as the selection of routes. To save energy, network broadcasts should be optimized. That is why, EOLSR consists of the four following modules [3]:

3.1.1. Energy Consumption Model

Energy efficient routing is aware of energy dissipated during a transmission. An energy consumption model is used by the routing protocol itself to take its decisions. It can also be used by the protocol

designers to evaluate the performances of their protocol. The energy consumed by a node depends on its state. In our work, we suppose that all nodes transmit with the same transmission power allowing to reach nodes at the maximum transmission range. As we are interested in the additional energy spent during the transmission of a flow packet from its source to its destination with reference to the idle state, the values used for transmission power and reception power are calculated as follows: $P_{trans} = P_{transmit} - P_{idle}$, $P_{recv} = P_{receive} - P_{idle}$. Now, we can determine the energy dissipated in transmitting (E_{trans}) or receiving (E_{recv}) a packet. Let $Duration$ denote the transmission duration of a packet. We have: $E_{trans} = P_{trans} * Duration$, $E_{recv} = P_{recv} * Duration$.

Interference and overhearing generate loss of energy. Hence, energy dissipated by a transmission by sender i is calculated as follows:

$$cost_{transmission}(i) = E_{trans} + n * E_{recv} \quad (1)$$

where n represents the number of non-sleeping nodes belonging to the overhearing and interference zones of the transmitter i . This cost indicates the quantity of energy consumed by a packet of the flow to reach the next hop. The energy dissipated by an end-to-end transmission over a path P can be computed as follows:

$$cost(flow) = \sum_{i \in sender(flow)} cost_{transmission}(i) \quad (2)$$

where i is a sender of $flow$ on its path P .

3.1.2. Energy Efficient Selection of MPRs

An energy efficient MPRs selection allows each node N in the network to select a subset of its one hop neighbors, according to their residual energy, that permits to reach any two-hop neighbor through a path with enough energy. The energy efficient MPR is named EMPR. We keep MPR to denote a MPR selected by the *OLSR* protocol.

To enable a selection of multipoint relays energy-aware, additional information about energy has to be included in the *Hello* message. We adopt the usual assumption that interferences are limited to two hops from the transmitter.

Let N be any node selecting its EMPRs. For any M one-hop neighbor of N , N computes $M1E(M)$, the weighted residual energy of M given by:

$$M1E(M) = \min\left(\frac{E_R(M)}{P_{trans} + P_{rcv}}, \min_{D \in 1hop(M)}\left(\frac{E_R(D)}{2 * P_{rcv}}\right)\right) \quad (3)$$

where $E_R(M)$ denotes the residual energy of node M . N sorts its one-hop neighbors by decreasing order of $M1E(M)$. Let $\mathcal{N}1(N)$ denote this ordered set. N loops around the following actions:

- if the first node in $\mathcal{N}1(N)$ covers at least one two-hop neighbor uncovered by the already selected EMPRs, then N selects this node as EMPR;
- N extracts this node from $\mathcal{N}1(N)$.

until all two-hop nodes are covered.

In the Equation 3, the weights of $E_R(M)$ and $E_R(D)$ take into account the role played by the nodes M (one-hop neighbor of N) and D (two-hop neighbor of N) in a transmission from N to D , via the

node M . They represent the maximum transmission duration that can be sustained. We can notice that the algorithm of EMPR selection tends to share the energy consumption between the different nodes. This selection avoids that nodes deplete their battery more quickly than others. However, with regard to the native MPR selection algorithm, this new algorithm introduces more frequent route changes because the residual energy of nodes decreases continuously. To limit these changes, we introduce two thresholds $ThresholdEMPR$ and $ThresholdMinEnergy$ and change the EMPR selection only if these two following conditions are met:

1. there is at least one 2-hop neighbor D such that: $\frac{E_R(new_EMPR) - E_R(old_EMPR)}{E_R(old_EMPR)} > ThresholdEMPR$, where $E_R(new_EMPR)$ represents the residual energy of the new node selected as EMPR to cover D , and $E_R(old_EMPR)$ represents the residual energy of the previous node selected as EMPR to cover D .
2. the residual energy of the new EMPR is sufficient: $E_R(new_EMPR) \geq ThresholdMinEnergy$. This avoids frequent changes when the residual energy of an EMPR tends to 0.

If these two conditions are met for at least one 2-hop neighbor, the previous set of EMPRs is changed by the new set. Each node computes its EMPRs before sending its periodic *Hello*.

3.1.3. Routing Algorithm for EOLSR

Our idea to maximize network lifetime, is to minimize the energy consumed by the end-to-end transmission of a flow in selecting the path, build from EMPRs, consuming the minimum energy. For that, we will use the cost given in the Equation 2 to compute routes for flows. Our target is to keep the complexity of Dijkstra (the same complexity as OLSR). However, instead of using the number of hops between source and destination to select the shortest route, as done in *OLSR*, we will use $cost(flow)$ as the criterion to choose the best path where: every link $i \rightarrow j$ has a cost equal to $cost_{transmission}(i)$ computed according to Equation 1. Network nodes are assumed to be in the idle state when the medium is idle, like in IEEE 802.11. In this case, the energy dissipated in receiving + overhearing + interferences takes into account the number of non-sleeping nodes in the 1-hop and 2-hop neighborhood of each sender of the considered flow as follows.

$$cost(flow) = \sum_{i \in sender(flow)} (E_{trans}(i) + \sum_{j \in \mathcal{N}_1(i) \cup \mathcal{N}_2(i)} E_{recv}(j)) \quad (4)$$

If several routes dissipate the same energy, the shortest one is chosen.

3.1.4. Optimized Broadcasts

In wireless networks, we usually need to broadcast messages to maintain network functionalities or advertise new services. However, flooding messages are very greedy on network resources. Consequently, optimizing the number of broadcast messages is very important and influences network performances. The idea is to use the MPRs nodes to forward broadcast messages and the EMPRs as intermediate nodes on routes. Hence, we keep the optimization of network broadcasts used in *OLSR* to optimize network flooding by means of the following forwarding rule:

A node forwards once a broadcast message with a non-null time-to-live only if it has received this message for the first time from a node that has selected it as MPR.

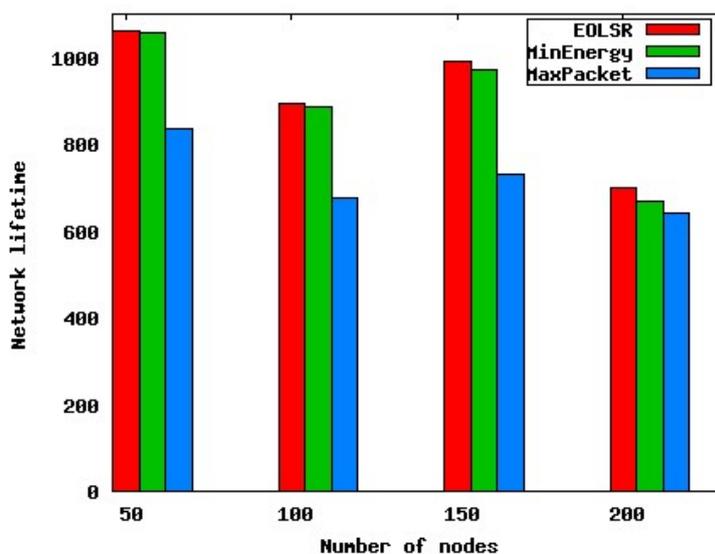
3.2. Performance Evaluation

We now evaluate the performances of EOLSR in various network configurations. Each configuration is characterized by a node number and a density. We recall that network density denotes the average number of neighbors per node + 1. For each configuration, we run 10 simulations and the result depicted in the curves is the average of these 10 simulations. Based on the simulation parameters given in Table 2, we compare the network lifetime obtained with EOLSR, and two other strategies:

- MinEnergy that selects the route consuming the minimum energy for an end-to-end transmission, without considering node residual energy. Notice that in this algorithm, the EMPRs of a node are its 1-hop neighbors that minimize the energy consumed to reach a 2-hop neighbor.
- MaxPacket that selects the routes that maximize the number of packets that can be transmitted from a source to its destination. It adopts the same selection of EMPRs as EOLSR but takes the route maximizing $\min_{i \in \text{sender}(\text{flow})} M1E(i)$.

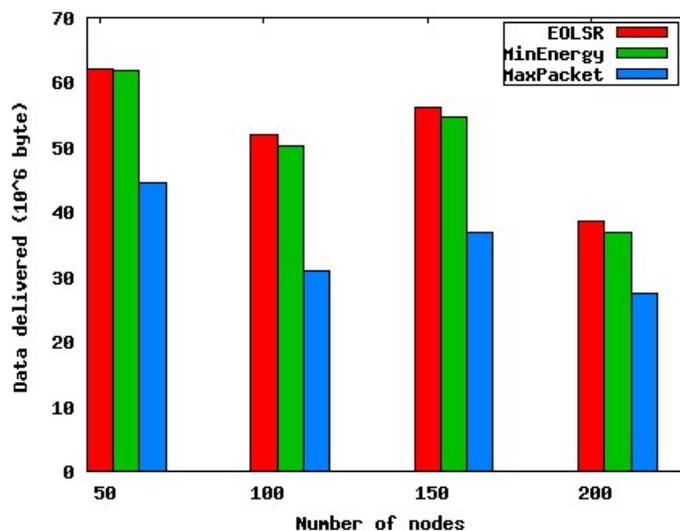
Simulation results concerning network lifetime [3] are illustrated in Figure 2. In all cases, MaxPacket provides the lowest network lifetime. This can be explained by the fact that the selected route does not minimize the energy consumed by an end-to-end transmission. MinEnergy does, but it always uses the same nodes: those belonging to the route of minimum energy. EOLSR that selects (1) the EMPRs according to their residual energy and this of their neighbors and (2) the routes of minimum energy built from these nodes, obtains better results.

Figure 2. Network lifetime obtained with EOLSR, MinEnergy and MaxPacket.



In this paper, we show that the increase in network lifetime is not a factice one. It really allows the application to transfer more useful data as depicted in Figure 3. These new simulation results justify both the selection of EMPRs according to Equation 3 and the selection of routes of minimum energy cost, two design choices of EOLSR.

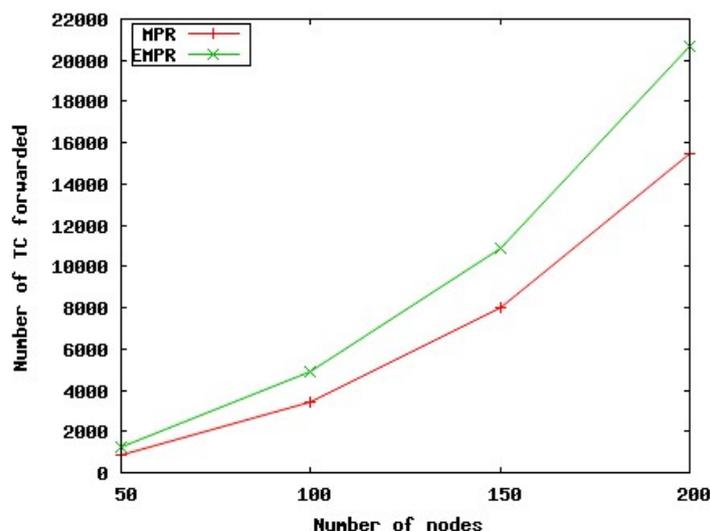
Figure 3. Comparison of the amount of data delivered with EOLSR, MinEnergy and MaxPacket.



In order to justify the choice of keeping MPRs nodes to optimize network broadcasts, we now evaluate the total number of TCs sent (cumulating generated TCs and forwarded TCs) during network lifetime in two cases:

- with EOLSR where TCs are forwarded by MPRs. This corresponds to the curve denoted MPR in Figure 4;
- with another solution where TCs are forwarded by EMPRs, solution denoted EMPR in Figure 4.

Figure 4. Number of TCs forwarded in the network.



We observe a significant improvement when MPRs, instead of EMPRs, are in charge of optimizing broadcasts. As the selection of EMPRs must take energy into account to maximize network lifetime, two types of multipoint relays are used in EOLSR: MPRs to optimize network broadcasts and EMPRs to build energy efficient routes. We can conclude from this performance evaluation that the simulation results validate our design choices made with EOLSR.

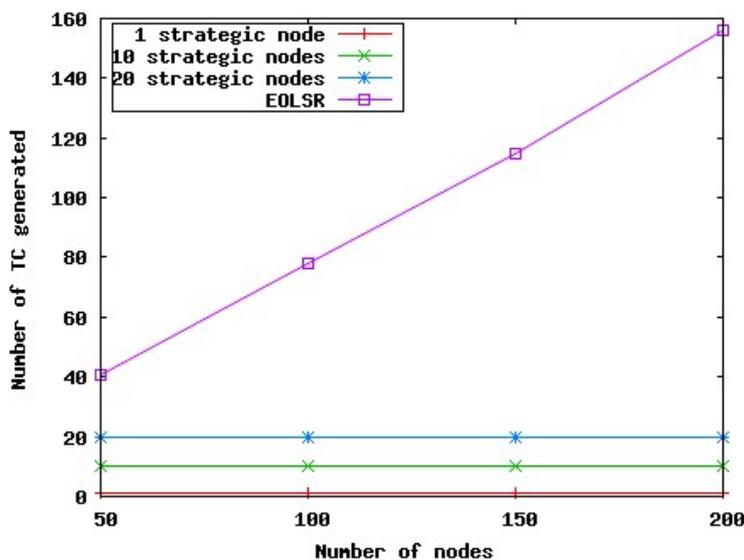
3.3. Cross Layering Optimization

3.3.1. Cross-layering with the Application Layer

In this paper, we investigate the use of cross-layering, taking advantage of the information provided by the application layer to reduce the overhead induced by *Hello* and *TC* messages. The simplest way consists in increasing the *Hello* and the *TC* periods. However, *Hello* messages are used to discover link breakages and link appearances; a higher period would result in an increase of protocol latency and a poor reactivity. A trade-off must be found. Some knowledge of the application can be very useful in selecting the relevant value of *Hello* and *TC* periods. For instance, we can define different phases for each application. It would be judicious to adapt the periods to each application phase. We can also notice that *Hellos* are sent only one-hop away, whereas *TCs* are sent all over the network. The use of MPRs allows to optimize broadcasts by reducing the number of *TC* retransmissions and usually exhibits better performance than other techniques [60].

Recall that *TCs* are sent to build the topology and the routing tables. We can mention that in some applications, like data gathering, it is useless to maintain a route to any node in the network. It is sufficient to maintain a route to the sink. That is why in EOLSR, two modes are distinguished: the general mode when no knowledge of the potential destinations is provided and the strategic mode when a few number of nodes is identified as potential destinations. These nodes are called strategic. A specific case is provided by data gathering applications with sink nodes.

Figure 5. Number of TCs generated in the network.



In the strategic mode, only strategic nodes periodically generate a *TC* message that is broadcast. A node receiving such a message will use the sender of this message as the next hop to the strategic node having generated this *TC*, if the energy dissipated to reach the strategic node (energy cost of the route) is less than the one recorded in its routing table. If so, the *TC* message is propagated. It follows that in a data gathering application, a sensor node needs to maintain only one route: the route to the sink. There is only one node generating *TC*: the sink. Figure 5 compares the number of TCs generated by EOLSR

in the generic mode and EOLSR in the strategic mode, assuming 1 strategic node, 10 strategic nodes or 20 strategic nodes. It appears that as long as the number of strategic nodes is less than the number of EMPRs, the strategic mode should be used. Notice that this is the usual case. With the strategic mode, the overhead is considerably reduced while meeting the application requirement: each node knows a route to reach any strategic node. In other words, this overhead reduction does not induce a degradation of the services provided to the application, but on the contrary improves the services by increasing the bandwidth part left to the application as well as the amount of energy left to application transmissions.

3.3.2. Cross-layering with the MAC Layer

Cross-layering with the MAC layer improves (1) EOLSR reactivity to topology changes by an early detection of broken links and (2) routes stability by the choice of good quality links. Indeed, the MAC layer provides the network layer with the received messages and their corresponding signal power. Using this information, EOLSR is able to select links with good quality and reject links with poor quality. Moreover, as soon as the received signal quality falls under a specific threshold, the corresponding link is immediately considered broken, instead of waiting for the expiration timer.

4. Node Activity Scheduling

We now present SERENA [10] that schedules node activity according to their color.

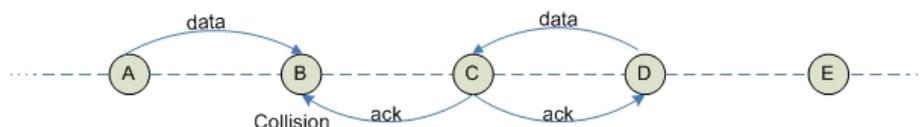
4.1. Justification of Design Choices

With regard to graph coloring, SERENA adopts three-hop vertex coloring. We now justify this choice.

4.1.1. Choice of Three-hop *versus* Two-hop Coloring

Three-hop coloring requires more colors than two-hop coloring and is more complex. However, it can be the only possible solution regarding the MAC layer choices. In wireless sensor networks, the wireless transmissions are subject to versatile radio propagation conditions and hence are not reliable. That is why, immediate acknowledgement of unicast transmissions is used at the MAC level. In this case, two-hop node coloring is not sufficient as illustrated in Figure 6. Nodes *A* and *D* have the same color. Node *C* immediately acknowledges *D* causing a collision on node *B*. If edge coloring is chosen, links should use the same color in both directions to support immediate acknowledgement. Two links *AB* and *CD* cannot have the same color if they are incident to the same node or to two one-hop neighbors. In a distributed implementation of edge coloring, a node colors its links. In such a case, a node must know the colors of links of any node up to 3 hops. This leads to a complexity higher than vertex coloring one.

Figure 6. Collision with two-hop coloring and immediate acknowledgement.



4.1.2. Choice of Vertex Coloring

Considering three-hop coloring, we now discuss the respective merits of edge and vertex coloring. Edge coloring can increase the concurrency of transmissions. In fact, if a color is assigned to a vertex then no one-hop, two-hop or three-hop neighbors can use the same color in a collision free assignment, assuming immediate acknowledgement. In Figure 7, where the node color is given by the integer near the node, only node *E* can have the same color as node *A*. With edge coloring, two three-hop neighbors can transmit or receive in the same slot if their corresponding nodes are at least two-hop away. For example in Figure 8, using edge coloring, where the link color is given by the integer on the link, node *A* can send a message to *B* while *D* three-hop away from *A* is sending a message to *E*. This is not possible with vertex coloring, where *D* three-hop away from *A* cannot have the same color as *A*, however it can receive data from *E* as illustrated in Figure 7.

Figure 7. Vertex coloring.

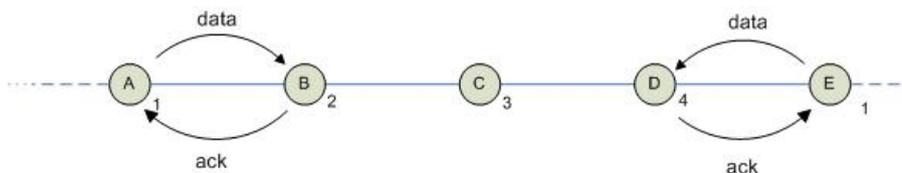
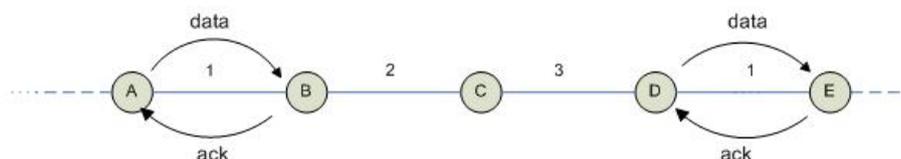


Figure 8. Edge coloring.



Edge coloring can assign a bandwidth amount proportional to the number of neighbors whereas with vertex coloring the assigned bandwidth is usually the same for all nodes. Edge coloring reduces energy lost in overhearing because only the sender and receiver are awake, unlike vertex coloring where all 1-hop neighbors of the transmitter are awake: they are potential receivers. However, if with edge coloring slots are not assigned carefully, a node can awake as many times as one-hop neighbors. Hence, the choice of contiguous slots would help in energy saving [61]. Moreover, edge coloring does not support broadcast communications. Nevertheless, broadcast messages are always needed for neighborhood discovery and route creation. Moreover, the MAC layer must maintain a buffer for each one-hop neighbor it has messages to send. Finally, the number of colors used to color all the edges of the network can be very important compared to the number of colors used to color all nodes in the network. That is due to the number of edges which is much greater than the number of nodes in the network. Consequently, delay can increase with the number of colors.

In this work, broadcast messages are used to manage the network and build the neighborhood. For this reason, we choose the vertex coloring approach.

4.2. Principles

In this paper, SERENA is specified as a generic solution able to adapt to various environment or architecture constraints as well as different application requirements such as:

- Broadcast transmission required or not.
- Types of unicast transmissions:
 - general: any node can transmit messages to any other node in the network;
 - tree-based: a node transmits a message either to its parent or to its children only.
- Immediate acknowledgement of unicast transmissions required or not. In case of immediate acknowledgement, the receiver uses the time slot granted to the sender to send its acknowledgement. Thus, the sender can retransmit immediately in case of unsuccessful receipt, as long as the maximum number of retries is not reached. Hence, immediate acknowledgement ensures shorter delivery delays, allows the sender to free its message quicker and avoids the receiver to store its acknowledgement in a queue before transmission.
- Minimizes the delay needed to collect or disseminate data in a tree or more generally in a hierarchical directed acyclic graph: a DAG where nodes are grouped into hierarchical levels (e.g., nodes that are at d hops from a given sink belong to level d).

In SERENA, any node N proceeds as follows:

- **Determination of $\mathcal{N}(N)$** , the set of nodes that cannot have the same color as N . This set is built according to the application constraints previously specified. For instance, in case of general communications with broadcast and without immediate acknowledgement, the set $\mathcal{N}(N)$ is the set of nodes up to two hops from N . If the immediate acknowledgement is supported, $\mathcal{N}(N)$ is the set of nodes up to three hops from N .
- **Computation of $priority(N)$** . The priority of node N is computed. It determines the order according which nodes are colored. In the case of general communications, it is equal to the cardinal of $\mathcal{N}(N)$, denoted $|\mathcal{N}(N)|$. In the case of a tree, it is equal to the number of descendants of N in the tree. The choice of the priority tends to minimize the number of colors needed.
- **Coloring rules:** they are two:
 - R1.** A node N colors itself if and only if it has the highest priority among the uncolored nodes in $\mathcal{N}(N)$.
 - R2.** When it colors itself, node N takes:
 - * the smallest color unused in $\mathcal{N}(N)$ in the general case,
 - * the smallest color unused in $\mathcal{N}(N)$ higher than the color of its parent in case of a tree or a hierarchical DAG structure.

Colors are represented by natural integers.

- During the coloring algorithm, **each node N sends periodically** a *Color* message to its one-hop neighbors. This message contains all the information needed to determine the set $\mathcal{N}(N)$, know the priority and color of any node in $\mathcal{N}(N)$ and detect message loss.

4.3. Performance Evaluation

We now evaluate by simulation the performance of SERENA three-hop vertex coloring in various network configurations. Similar to EOLSR, we run 10 simulations per configuration and the result depicted in the curves is the average of these 10 simulations. Simulation parameters are those given in Table 2. Figure 9 depicts the number of colors needed when varying network density and number of nodes, like in [10]. We notice that the curves depicting the number of colors as a function of the node number are always under the first bisectrix. In other words, the number of colors is always smaller than the node number. Higher the difference, higher the benefits provided by coloring. The number of colors strongly increases with density and weakly with the node number. Compared with two-hop coloring as evaluated in [6], three-hop coloring requires about 50% additional colors. For instance, to color 200 nodes with a density 10, 28 colors are sufficient with two-hop coloring, whereas 43.4 colors are needed with three-hop coloring. However, if immediate acknowledgement is required, three-hop coloring is needed to avoid the color conflicts depicted in Figure 6.

Figure 9. Number of colors used by SERENA for different network configurations.

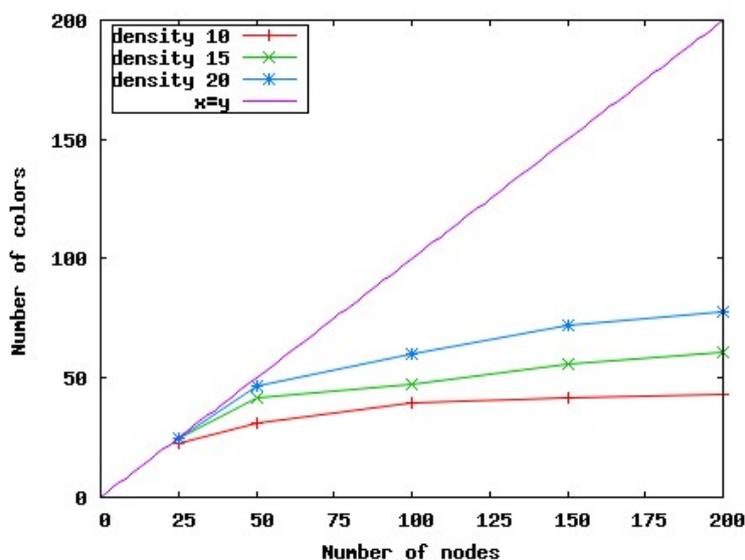


Figure 10 shows the number of rounds needed to color all network nodes. It represents the complexity of the coloring algorithm. This number of rounds is close to the number of nodes, whereas the worst theoretical case gives an upper bound of three times the number of nodes for three-hop coloring. In other words, after one node has colored itself, the only node that can color next is three-hop away. Such a configuration is unfrequent in usual wireless sensor networks in industrial environment.

The average number of messages sent per node is given in Figure 11 for different densities and numbers of nodes. We observe that the number of messages like the number of colors strongly depends on network density and weakly with the number of nodes. Simulation results show that the number of messages generated by three-hop coloring is about twice the number generated by two-hop coloring. This is due to the fact that information about 1, 2 and three-hop neighbors is needed to color a node. These new results allow us to conclude that the immediate acknowledgement is expensive both in terms of

colors and messages. However, it is needed to quickly confirm the good reception of a unicast message, which is not guaranteed in a wireless environment.

Figure 10. Number of rounds used by SERENA for different network configurations.

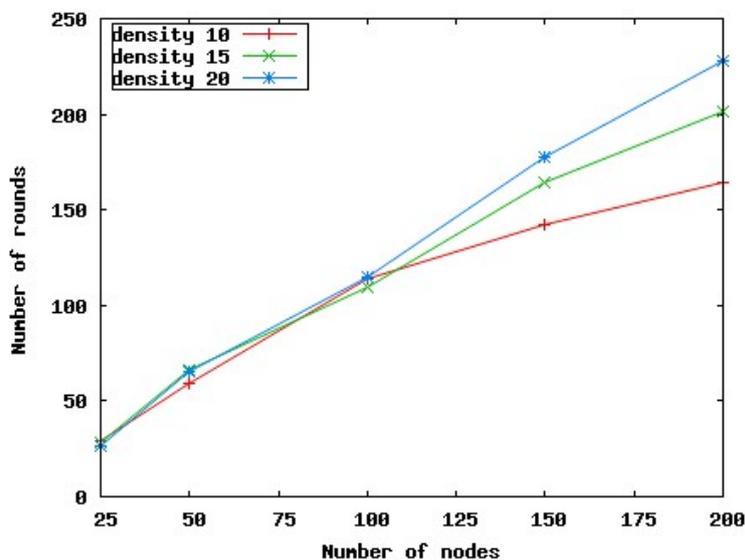
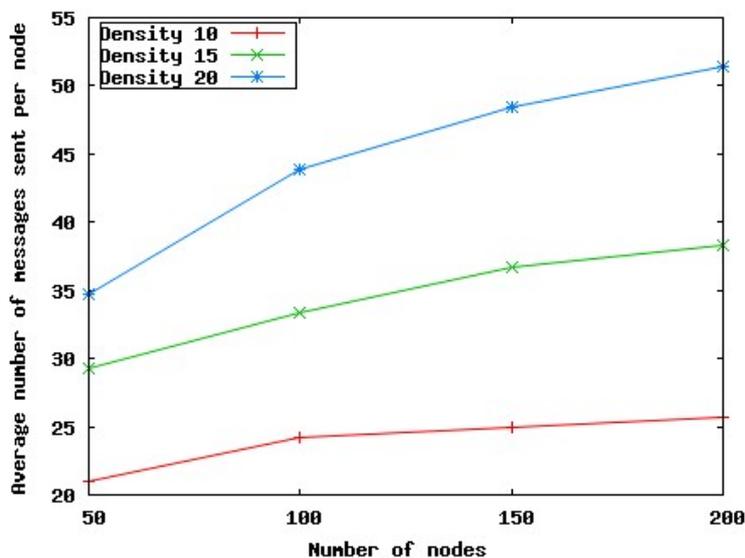


Figure 11. Average number of messages sent per node.



4.4. Cross Layering Optimization

Some knowledge of the application and MAC layer can improve the performances of SERENA.

4.4.1. Cross-layering with the Application Layer

The knowledge provided by the application layer can help SERENA to determine when the node can sleep. Indeed, we have designed a specific functioning mode for SERENA which targets data gathering applications. The idea is to determine a color assignment order such that no child is scheduled after its

parent in an upstream communication. Thus, a node can aggregate data received from its children before transmitting them to its parent. Consequently, the delay needed by the sink to collect data from all sensor nodes is minimized.

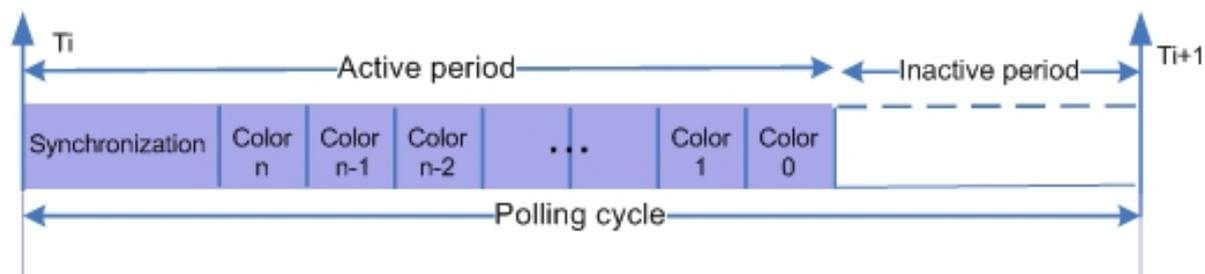
As mentioned in Section 4.2, a node selects the smallest color available in $\mathcal{N}(N)$ higher than the color of its parent. Consequently:

1. the first node to color itself is the root of the tree.
2. each node has a color strictly higher than the color of its parent in the tree.

The choice of the color in a tree is such that all children of any node N transmit before N in case of data gathering, or after N in case of data dissemination. This contributes to considerably reduce the end-to-end delays. Indeed, in a single cycle, all data can be collected from the sensors and delivered to the sink, assuming data aggregation. Similarly, in a single cycle, data broadcast by the sink reach all network nodes. To achieve that, it suffices to schedule node transmission in the decreasing order of colors for data gathering and in the increasing order of colors for data dissemination. That is the role of the MAC layer.

The MAC cycle illustrated in Figure 12 comprises an activity period followed by an inactivity period during which all nodes sleep. The activity period consists in the synchronization period and color slots. During a color slot, only nodes having this color are allowed to transmit data.

Figure 12. MAC cycle.

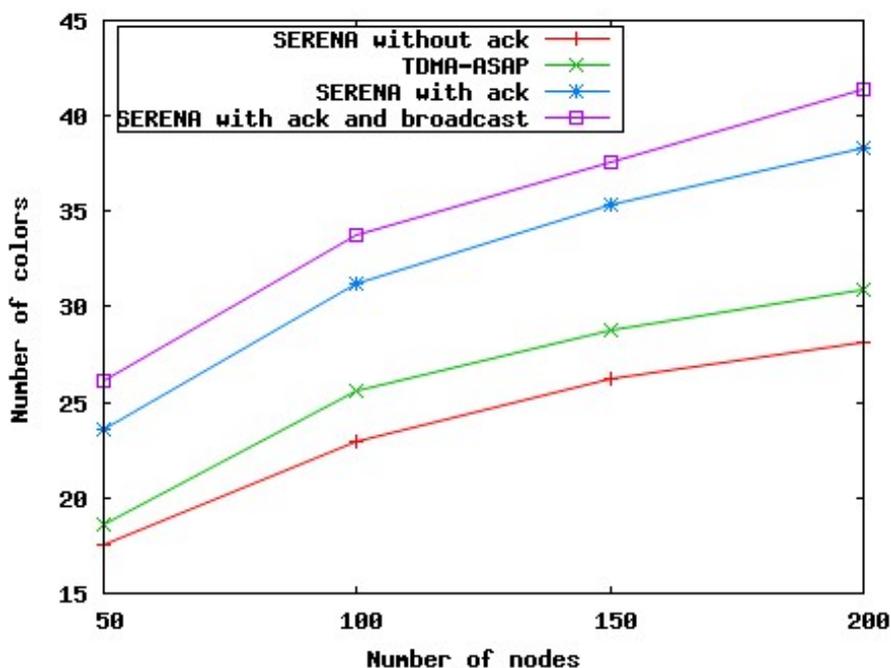


We can also observe that with SERENA tree-coloring, there are two types of cycle: one favoring the downstream traffic (*i.e.* starting with color 0 of the sink) and the other favoring the upstream traffic (*i.e.* starting with the maximum color used in the network). The frequency of each cycle is determined by the application traffic.

Figure 13 allows us to compare SERENA with TDMA-ASAP [36], when communications are done according to a tree. Hence, with SERENA the color of a node is higher than the color of its children. We consider various configurations: the node number ranges from 50 to 200 with a network density of 10. We evaluate three variants of SERENA depending on application requirements:

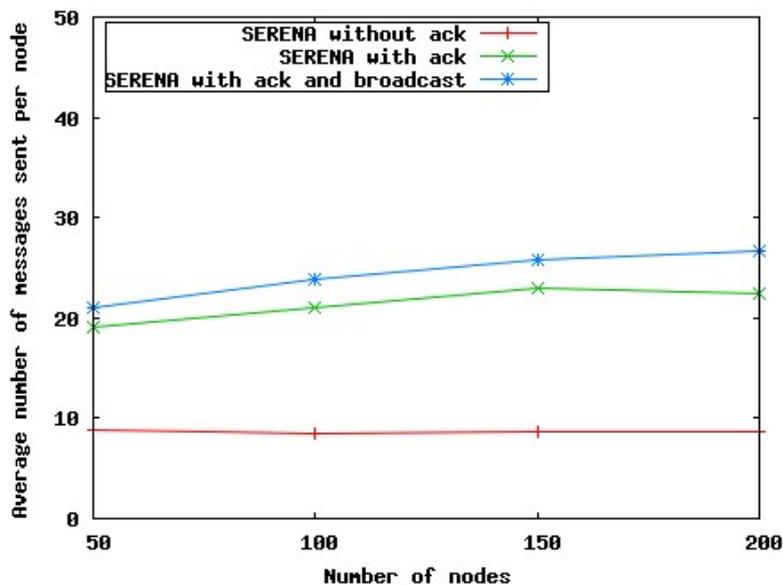
- without immediate acknowledgement and without broadcast,
- with immediate acknowledgement and without broadcast,
- with immediate acknowledgement and with broadcast.

Figure 13. Comparison of the number of colors used in SERENA and TDMA-ASAP.



This allows us to quantify the cost in additional colors induced by the immediate acknowledgement and the broadcast. The cost of the immediate acknowledgement is higher than the cost of broadcast. For example, in a network with 100 nodes and a density of 10, 8 additional colors are needed for the immediate acknowledgement and only 3 for the broadcast. It follows that for a density 10, the support of immediate acknowledgement costs about 35% of additional colors, whereas the support of broadcast costs less than 10% of additional colors; this is true for 100 nodes as well as 200 nodes. TDMA-ASAP colors the tree, level by level, starting with the farthest level from the sink (*i.e.*, the reverse of SERENA). A node receives a color strictly higher than the color of its parent. Two nodes having the same parent do not share the same color. Moreover, a node that is neighbor of the parent of another node does not share its color. With these coloring rules, TDMA-ASAP uses a number of colors smaller than the two variants of SERENA with immediate acknowledgement. That is because TDMA-ASAP does not support immediate acknowledgement. Our variant of SERENA without immediate acknowledgement performs better than TDMA-ASAP, because the node priority in SERENA coloring algorithm is equal to its number of descendants. Thus, nodes with a higher number of descendants tend to color first, taking smaller colors and then reducing the constraints on the colors taken by their descendants.

Figure 14 depicts the average number of messages sent by a node during the coloring process of a tree with regard to different application requirements. TDMA-ASAP has not been represented because the coloring algorithm given in [36] is centralized. It turns out that for 100 nodes the additional number of color messages due to the immediate acknowledgement is about 148%, whereas the additional number of color messages due to broadcast is limited to about 13%. Notice that this additional cost is paid only when the coloring is performed.

Figure 14. Average number of messages sent per node in a tree.

4.4.2. Cross-layering with the MAC Layer

SERENA provides the MAC layer with the color of each node, the color of its one-hop neighbors and the maximum number of colors used. Given this information, the MAC layer dimensions the frame, assigns active time slot to each node according to its color. During a time slot, all the nodes having the corresponding color can transmit their messages whereas their one-hop neighbors must remain awake to receive them. The other nodes can sleep. The medium access is scheduled in such a way that two nodes with the same color can transfer simultaneously without interfering. No energy, no bandwidth is lost in collisions.

However, in a real wireless environment, because of the existence of unidirectional links, node mobility, late arrivals of nodes or instability of radio propagation conditions, colors conflict can occur. In this case, the MAC layer is the first layer able to detect this conflict. Hence, it can inform SERENA of conflicting nodes. SERENA solves this conflict by assigning a new color to one conflicting node [10].

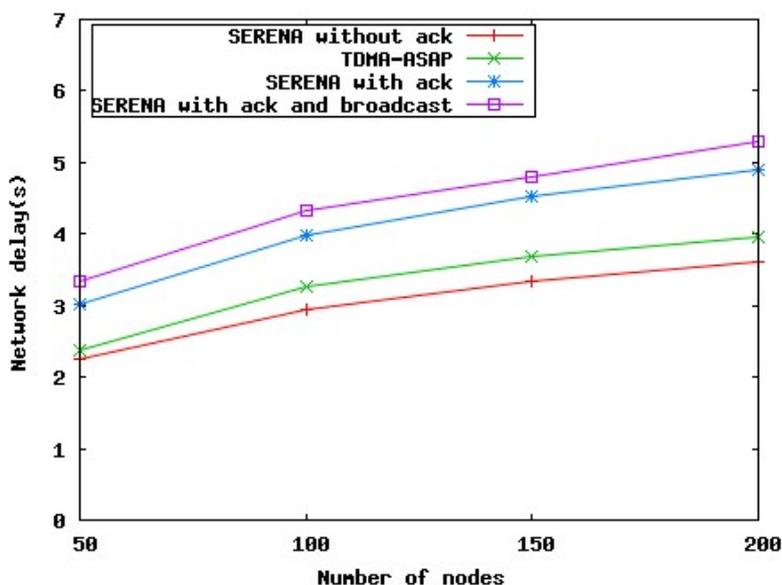
4.4.3. End-to-end Delays and Reliability

Let us now evaluate the end-to-end delays in a data gathering application, assuming a MAC cycle given in Figure 12. Figure 15 depicts the network delays, assuming a slot duration of 128 ms. The network delay represents the time elapsed from the time where the farthest node from the sink starts to transmit to the time where this message is received by the sink. We then have: $network\ delay = number\ of\ colors \cdot slot\ duration$. For the evaluation illustrated in Figure 15, each node has 30 bytes to transmit. Packet size is 128 bytes.

For configurations with 200 nodes, we assume that any node other than the sink has at most 100 descendants in the tree. Assuming a slot duration of 128 ms, we get an active period of 5.376 s for 200 nodes and 42 colors with SERENA + immediate acknowledgement + broadcast. Notice that the slot duration can be adapted to the maximum number of descendants of the children of the sink. For 100 nodes and assuming a slot duration of 64 ms and any node other than the sink having at most 50

descendants in the tree, a slot duration of 64 ms, we get an active period of 2.176 s and 34 colors with SERENA + immediate acknowledgement + broadcast. We observe that the network delay curves are very similar to the number of colors curves. This is expected because of the formula giving the network delay from the number of colors.

Figure 15. Network delays in a tree.

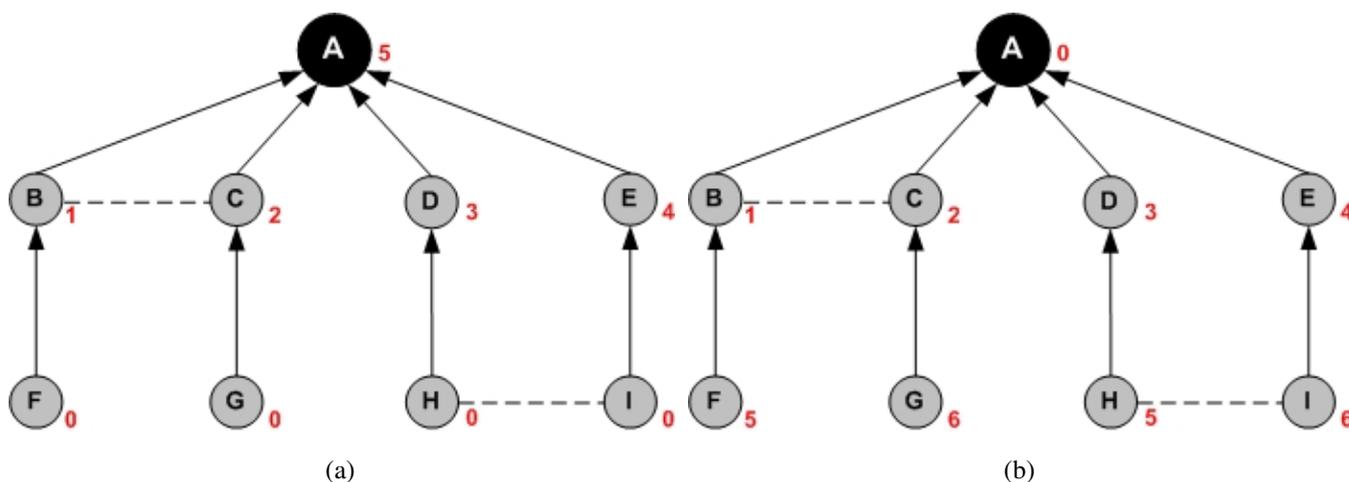


We can also compute the maximum end-to-end delays, considering that the farthest node from the sink wants to transmit just at the end of its color slot. It must then wait until the next cycle. Hence, the maximum end-to-end delay is equal to the network delay increased by the cycle duration. Notice that without tree coloring, a message from the farthest node can take a number of cycles equal to 1+ its distance in hop number to the sink. Hence, tree coloring brings a huge benefit with regard to end-to-end delays.

If now we focus on reliability, we observe that the inherent versatility of the wireless environment makes the transmissions unreliable: message losses are possible. To cope with them, two solutions are possible. Either sampling at a frequency higher than required in order to tolerate message losses. Or detecting losses by using acknowledgements: immediate or delayed. As already said, immediate acknowledgements are easier to implement in the MAC layer and lead to shorter transmission delays. That is why they are frequently used. If we use TDMA-ASAP that has not been designed to support this immediate acknowledgement, potential color conflicts can occur. They correspond to situations where two nodes having the same color transmit simultaneously, causing a collision between a data frame and an acknowledgement frame and making impossible for the destination to receive its intended frame. Such conflicts cannot be avoided by the capture effect. The number of such situations depend on the tree and the network topology. To illustrate our purpose, we take a very simple example with 9 nodes. The topology is depicted in Figure 16, where circled letters denote nodes. A plain line denotes a link in the tree whereas a dotted line denotes a neighbor link unused in the tree. The number besides the circled letter denotes the color granted to this node. We apply two tree colorings to this topology: TDMA-ASAP and SERENA with immediate acknowledgement. Figure 16(a) shows that with TDMA-ASAP, 6 colors

are needed. All the leaves have the same color. Notice also that this coloring does not support broadcast: nodes H and I broadcast simultaneously, each of them will be unable to receive the message of the other. With SERENA, as shown in Figure 16(b), 7 colors are needed, broadcast being required or not. This additional color is needed to avoid any color conflict due to the immediate acknowledgement. With TDMA-ASAP coloring, we count two potential color conflicts. Thus, nodes F and G cannot share the same color without causing a conflict between the data sent by F and the acknowledgement sent by C to G (respectively the data sent by G and the acknowledgement sent by B to F). Similarly, nodes H and I cannot share the same color without causing a conflict between the data sent by H and the acknowledgement sent by E to I (respectively the data sent by I and the acknowledgement sent by D to H). The number of occurrences of these conflicts during network lifetime strongly depends on the traffic generated by the application. With SERENA, there is no such conflicts, no collision occurs.

Figure 16. Example of tree coloring: (a) TDMA-ASAP coloring; (b) SERENA with immediate acknowledgement coloring.



5. Integration of EOLSR and SERENA

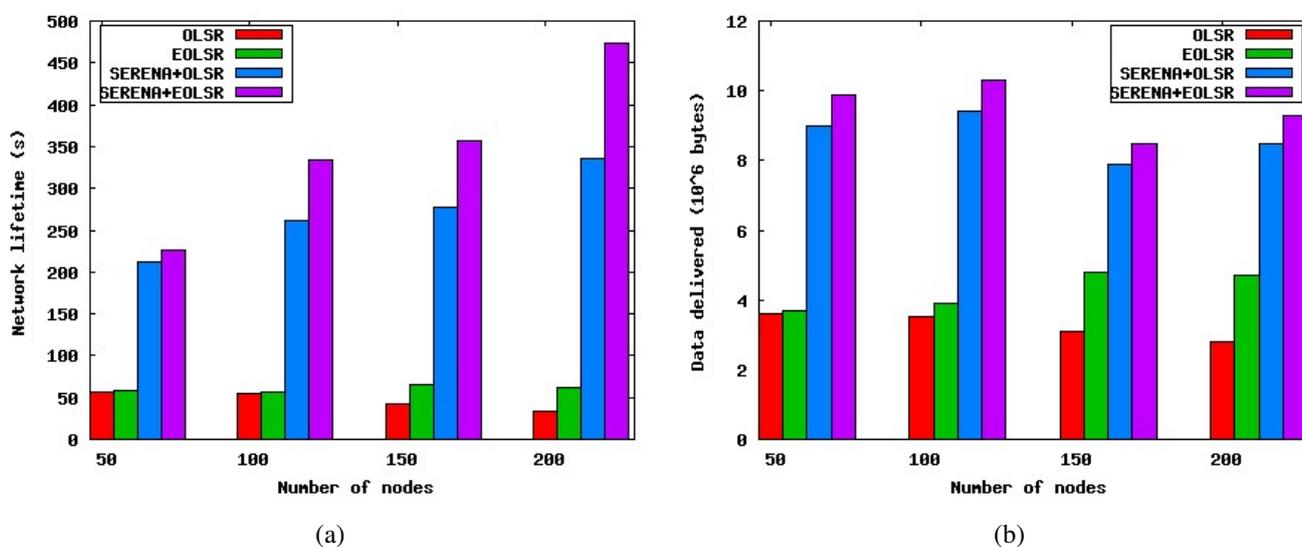
5.1. Benefits Resulting from the Integration

Integrating EOLSR with SERENA requires the choice of a coloring supporting broadcast. Indeed, *Hello* and *TC* messages of EOLSR are broadcast. Thus, for general communications with immediate acknowledgement, three-hop coloring is needed. For tree communication with immediate acknowledgement, “SERENA with ack and broadcast” is required. These *Hello* and *TC* messages must be taken into account in the dimensioning of the color slot duration.

We now evaluate the benefits brought by the integration of EOLSR and SERENA (three-hop coloring). We first quantify the increase of network lifetime brought by EOLSR, SERENA+OLSR and SERENA+EOLSR respectively in different network configurations. Network density is set to 10, the number of nodes ranges from 50 to 200. In these simulations, the *Hello* and *TC* messages have not been considered in order to get simulation results more quickly. We observe in Figure 17(a) [8] that compared to OLSR, EOLSR improves network lifetime, the improvement is more noticeable when the number of nodes is higher than 100. This can be explained by a higher number of possible routes. SERENA

provides the most important contribution in network lifetime increase. For instance, with 100 nodes, the network lifetime is multiplied by 6 compared to OLSR. Furthermore, the integration of EOLSR and SERENA leads to the highest network lifetime. We point out that this lifetime increase is higher than the sum of the increases obtained by EOLSR and SERENA taken separately. As a new result, we evaluate the amount of data delivered to the user. We observe that this lifetime increase is accompanied by an increase in the amount of user data delivered as shown in Figure 17(b). This improvement represents a real benefit of the SERENA+EOLSR integration as it is directly observable by the end user.

Figure 17. Comparative performances of OLSR, EOLSR, SERENA+OLSR, SERENA+EOLSR. (a) Network lifetime; (b) Amount of user data delivered.

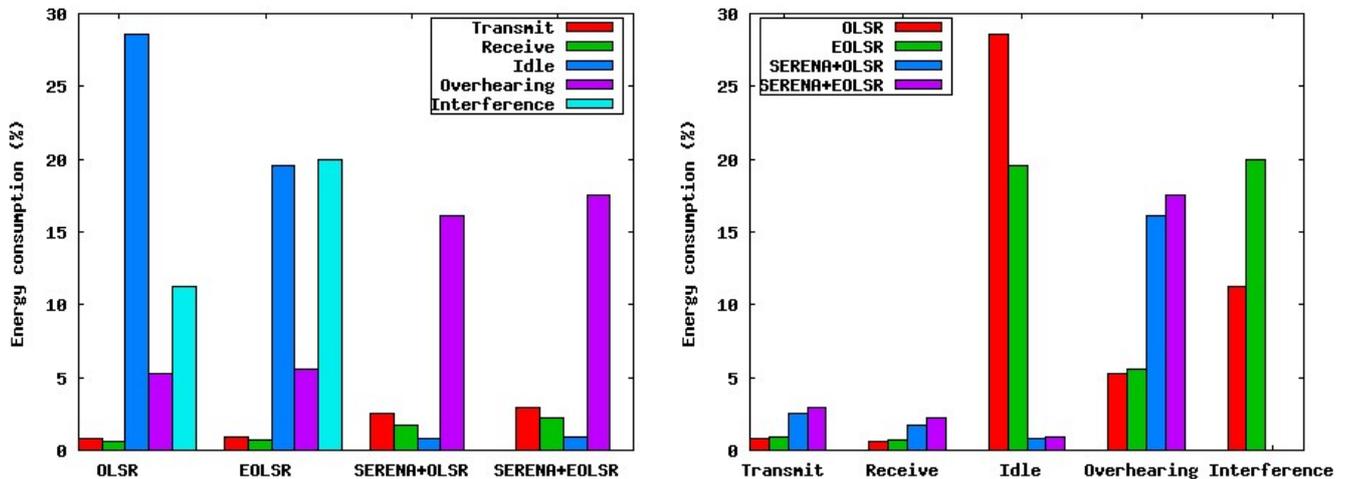


We now quantify the distribution of energy consumed in the different radio states for OLSR, EOLSR, SERENA+OLSR and SERENA+EOLSR. Results are illustrated in Figure 18 for a network of 100 nodes and a density of 10. These results extend those published in [7] by considering three-hop coloring instead of two-hop coloring.

Figure 18 determines the energy consumed by each protocol in each radio state: the left part of the figure highlights the protocol whereas the right part highlights the state. Unsurprisingly, OLSR has a poor energy efficiency: the main part of energy is consumed in the idle and interference states. The useful states, transmit and receive, represent the smallest part of consumed energy. EOLSR decreases the energy part consumed in the idle state. This is because the EMPRs selection takes into account the residual energy of nodes and avoids nodes with a low residual energy. In other words, EOLSR tends to balance the residual energy of nodes by changing the EMPRs selection when the energy of these nodes significantly decreases, unlike OLSR where the same intermediate nodes are in charge of transmitting. Hence, with EOLSR the number of nodes that have transmitted during network lifetime is higher. This, combined with a longer network lifetime, increases the energy lost in interferences. Another striking result is that no energy is lost in interferences with SERENA, leaving more energy to send and receive user messages. This can be explained by the three-hop vertex coloring: the nodes two-hop away from a transmitter are sleeping. As expected, SERENA considerably reduces the energy wasted in the idle

state. The integration of SERENA and EOLSR leads to a better energy efficiency: more energy is used to transmit and receive user messages.

Figure 18. Comparison of the energy consumption in OLSR, EOLSR, OLSR+SERENA, EOLSR+SERENA.



This performance evaluation shows that SERENA+EOLSR together contribute to a more efficient use of network resources. Node coloring avoids collisions. Bandwidth is spatially reused. No energy is lost in interferences. The use of EMPRs contributes to balance energy consumption. Routes selected by EOLSR tend to minimize the energy consumed by an end-to-end transmission.

5.2. Application to the OCARI Project

In the OCARI project, devoted to wireless sensor networks in an industrial environment, SERENA and EOLSR belong to the network layer of the OCARI stack illustrated in Figure 19.

This OCARI stack refines this published in [7]. EOLSR and SERENA interact with MaCARI layer via two service access points: MDE-SAP for MaCARI data and MME-SAP for MaCARI management.

In OCARI, the medium access is scheduled in three periods:

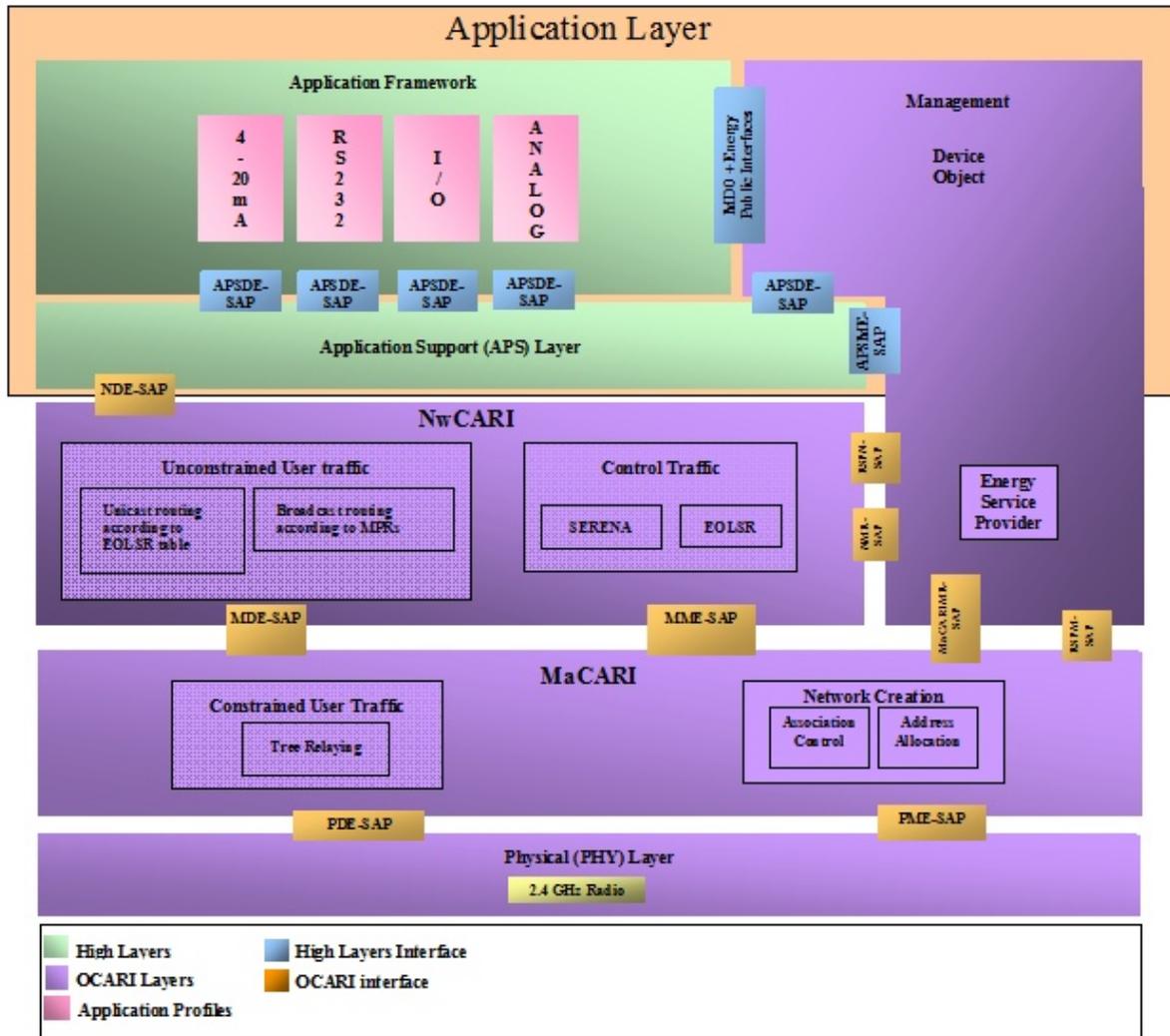
- $[T_0, T_1]$ for the synchronization: the beacon generated by the PAN coordinator is propagated multihop in the network;
- $[T_1, T_2]$ for the intra-cell communications: a cell consists of a coordinator and its attached nodes with reduced functionality. In this period, constrained and unconstrained traffic sent by a reduced functionality device is received by its coordinator and vice-versa;
- $[T_2, T_3]$ for the inter-cell communications: coordinators route unconstrained traffic using EOLSR.

It is possible to have an inactivity period $[T_3, T_0]$ during which all nodes sleep.

More precisely, the EOLSR routing protocol is used to route unconstrained traffic in the $[T_2, T_3]$ period devoted to inter-cell communications. As said in Section 3, EOLSR builds a routing table according to the information exchanged in the *Hello* and *TC* messages. This routing table is read by the network layer to find the next hop for the final destination of unconstrained traffic. EOLSR is able to find the path consuming the least energy from a source to a destination, even in case of mobility

of some nodes. The periodic exchange of *Hellos* will detect the appearance of new links as well as the breakage of existing ones. *TC* will propagate the topology changes in the network and routes are updated accordingly.

Figure 19. The OCARI stack.



SERENA is also used in this $[T2, T3]$ period to schedule coordinators activity. Coordinators run the SERENA coloring algorithm. The MaCARI layer of the PAN coordinator is informed of the number of colors used. It dimensions the frame accordingly and assigns time slots per color. Hence, the MaCARI layer of each coordinator is in charge of awaking it in the slot assigned to its color as well as in the slots assigned to its one-hop neighbors. In a perfect environment without mobility and late node arrivals, all transmissions would be collision free and then energy is saved. The medium access is perfectly deterministic. However, in a real industrial environment, color conflicts can occur. Two nodes with the same color transmitting simultaneously can cause collision because of a modified neighborhood due to late arrivals, mobility. Thus, these two nodes that were neither one-hop, nor two-hop nor three-hop neighbors can become neighbors and then prevent the intended destination of a message to correctly receive it. In OCARI, cross-layering between MaCARI and SERENA allows the detection/resolution of such conflicts. MaCARI will detect the conflict and notify SERENA of the conflicting nodes. SERENA

will solve the conflict by obliging the conflicting node with the smallest priority to change its color. For more details, see the paper by M. Misson *et al.* in this issue.

In a further step, SERENA will also be used in the $[T_0, T_1]$ period to reduce the time needed for beacon dissemination following the association tree. That is because nodes sharing the same color can transmit simultaneously without interfering and each node has a color strictly higher than the color of its parent in the tree. Hence, assigning time slots in the increasing order of colors reduces the end-to-end delay of downstream communications like beacon dissemination. Whereas the reverse order favors upstream communications. This assignment can change to alternatively favor upstream or downstream communications depending on the application requirements. Furthermore, the use of SERENA in the $[T_1, T_2]$ period will allow several cells to be simultaneously active without interfering. With SERENA, bandwidth is saved because of collision avoidance and spatial reuse. More generally, network resources are more efficiently used.

A prototype integrating OCARI physical layer, MaCARI, SERENA and EOLSR is being implemented. It shows the feasibility of the OCARI stack and will be experimented in an industrial environment.

6. Conclusions

Energy efficient routing and node activity scheduling in wireless sensor networks constitute a challenging research area. The OCARI project aims at designing and developing solutions for an industrial environment. The proposed solutions include EOLSR and SERENA. EOLSR selects routes that minimize the energy consumed by an end-to-end transmission while avoiding nodes with low residual energy. SERENA schedules node activity based on node coloring. It is a generic algorithm able to adapt to specific application requirements, environment constraints and various types of communications. For instance, to support immediate acknowledgement, often needed in an unreliable wireless communication, three-hop coloring is required. Simulation results show that both SERENA and EOLSR contribute to a more efficient use of network resources and increase network lifetime while delivering a higher amount of user data. Benefits resulting from the integration of EOLSR and SERENA are higher than the sum of the benefits obtained by each separately.

Cross-layering with the application layer significantly contributes to improve the performances of both SERENA and EOLSR. For instance, in data gathering applications, EOLSR uses the knowledge of potential destinations to considerably reduce the overhead. Similarly, SERENA is able to reduce the end-to-end delays in such applications, adapting the coloring rules to a tree topology or a hierarchical DAG.

A prototype integrating the physical layer, MaCARI, SERENA and EOLSR is under implementation. We will collect measures made on this prototype and compare them with other solutions, before experimenting our solution in a larger industrial network.

Acknowledgements

This work has been partly funded by the ANR OCARI project.

References

1. Mahfoudh, S.; Minet, P. An Energy Efficient Routing based on OLSR in Wireless *Ad Hoc* and Sensor Networks. In Proceedings of PAEWN 2008: IEEE International Workshop on Performance Analysis and Enhancement of Wireless Networks, Ginowan, Japan, March 2008.
2. Mahfoudh, S.; Minet, P. EOLSR: An energy efficient routing protocol in wireless *ad hoc* and sensor networks. *JOIN* **2008**, *9*, 389–408.
3. Mahfoudh, S.; Minet, P. Energy-aware routing in wireless *ad hoc* and sensor networks. In Proceedings of IWCMC 2010: 6th International Wireless Communications and Mobile Computing Conference, Caen, France, June 2010.
4. Minet, P.; Mahfoudh, S. SERENA: SchEduing RoutEr Nodes Activity in wireless *ad hoc* and sensor networks. In Proceedings of IWCMC 2008: IEEE International Wireless Communications and Mobile Computing Conference, Crete Island, Greece, August 2008.
5. Mahfoudh, S.; Minet, P. Performance evaluation of the SERENA algorithm to SchEdule RoutEr Nodes Activity in wireless *ad hoc* and sensor networks. In Proceedings of AINA 2008: IEEE 22nd International Conference on Advanced Information Networking and Applications, Ginowan, Japan, March 2008.
6. Mahfoudh, S.; Minet, P. Maximization of energy efficiency in wireless *ad hoc* and sensor networks with SERENA. *J. Mob. Inform. System.* **2009**, *5*, 33–52.
7. Dang, T.; Val, T.; Guitton, A.; Al Agha, K.; Minet, P.; Bertin, M-H.; Viollet, J-B. Which wireless technology for industrial wireless sensor network? The development of OCARI technology. *IEEE Trans. Ind. Electron.* **2009**, *56*, 4266–4278.
8. Bertin, M.-H.; Van Den Bossche, A.; Chalhoub, G.; Dang, T.; Mahfoudh, S.; Rahme, J.; Viollet, J.-B. In Proceedings of OCARI for industrial wireless sensor networks, IFIP Wireless Days, Dubai, UAE, November 2008.
9. Al Agha, K.; Chalhoub, G.; Guitton, A.; Mahfoudh, S.; Minet, P.; Misson, M.; Rahme, J.; Val, T.; Van Den Bossche, A. Cross-layering in an industrial wireless sensor network: Case study of OCARI. *J. Network.* **2009**, *4*, 411–420.
10. Minet, P.; Mahfoudh, S.; Chalhoub, G.; Guitton, A. Node coloring in a wireless sensor network with unidirectional links and topology changes. In Proceedings of IEEE WCNC 2010, Sidney, Australia, April 2010.
11. Kulik, J.; Rabiner, W.; Balakrishnan, H. Adaptive protocols for information dissemination in wireless sensor networks. In Proceedings of 5th ACM/IEEE Mobicom Conference, Seattle, WA, USA, August 1999.
12. Intanagonwiwat, C.; Govindan, R.; Estrin, D. Directed Diffusion: A scalable and robust communication paradigm for sensor networks. In Proceedings of MobiCom 2000, Boston, MA, USA, August 2000.
13. Heinzelman, W.; Chandrakasan, A.; Balakrishnan, H. Energy-efficient communication protocol for wireless microsensor networks. In Proceedings of HICSS'00, Maui, HI, USA, January 2000; Volume 2, pp. 3005–3014.

14. Manjeshwar, A.; Agrawal, D.P. TEEN: A protocol for enhanced efficiency in wireless sensor networks. In Proceedings of Workshop on Parallel and distributed Computing Issues in Wireless Networks and Mobile Computing, San Francisco, CA, USA, April 2001.
15. Manjeshwar, A.; Agrawal, D.P. APTEEN: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks. In Proceedings of Workshop on Parallel and distributed Computing Issues in Wireless Networks and Mobile Computing, Fort Lauderdale, FL, USA, April 2002.
16. Yu, Y.; Govindan, R.; Estrin, D. *Geographical and Energy-aware Routing: A Recursive Data Dissemination Protocol for Wireless Sensor Networks*; UCLA-CSD TR-01-0023; University of California, Los Angeles: Los Angeles, CA, USA, 2001.
17. Xu, Y.; Heidemann, J.; Estrin, D. Geography-informed energy conservation for *ad hoc* routing. In Proceedings of MobiCom 2001, Rome, Italy, July 2001.
18. Allard, G.; Minet, P.; Nguyen D.Q.; Shrestha, N. Evaluation of the energy consumption in MANET. In Proceedings of Adhoc-Now 2006, Ottawa, Canada, August 2006.
19. Hassanein, H.; Luo, J. *Reliable Energy Aware Routing in Wireless Sensor Networks*; IEEE Computer Society: Washington, DC, USA, 2006.
20. Shah, R.C.; Rabaey, J.M. Energy Aware Routing for Low Energy *Ad Hoc* Sensor Networks. In Proceedings of IEEE WCNC 2002, Orlando, FL, USA, March 2002.
21. Nasipuri, A.; Das, S. On-demand multipath routing for mobile *ad hoc* networks. In Proceedings of International Conference on Computer Communications and Networks, Boston, MA, USA, October 1999.
22. Cardei, M.; Du, D. Improving wireless sensor network lifetime through power aware organization. *ACM J. Wirel. Netw.* **2005**, *11*, 333–340.
23. Cardei, M.; Thai, M.; Li, Y.; Wu, W. Energy-efficient Target Coverage in Wireless Sensor Networks. In Proceedings of IEEE INFOCOM 2005, Miami, FL, USA, March 2005.
24. Carle, J.; Simplot-Ryl, D. Energy-efficient Area Monitoring for Sensor Networks. *Comput. Mag.* **2004**, *37*, 40–46.
25. Ye, W.; Heidemann, J.; Estrin, D. An Energy-Efficient MAC Protocol for Wireless Sensor Networks. In Proceedings of IEEE INFOCOM, New York, NY, USA, June 2002.
26. Dam, T.; Langendoen, K. An adaptive energy-efficient MAC protocol for wireless sensor networks. In Proceedings of ACM SenSys'03, Los Angeles, CA, USA, November 2003.
27. Lu, G.; Krishnamachari, B.; Raghavendra, C.; An Adaptive Energy-efficient and Low-latency MAC for Data Gathering in Wireless Sensor Networks. In Proceedings of Parallel and Distributed Processing Symposium, Santa Fe, NM, USA, April 2004.
28. Lu, G.; Krishnamachari, B.; Raghavendra, C. O-MAC: An Organized Energy-aware MAC Protocol for Wireless Sensor Networks. In Proceedings of IEEE ICC, Glasgow, UK, June 2007.
29. Krumke, S.; Marathe, M.; Ravi, S. Models and approximation algorithms for channel assignment in radio networks. *Wirel. Netw.* **2001**, *7*, 575–584.
30. Ramaswami, R.; Parhi, K. Distributed scheduling of broadcasts in a radio network. In Proceedings of INFOCOM 1989, Ottawa, Canada, April 1989.

31. Chakraborty, G. Genetic algorithm to solve optimum TDMA transmission schedule in broadcast packet radio networks. *IEEE Trans. Commun.* **2004**, *52*, 5.
32. Gandham, S.; Dawande, M.; Prakash, R. Link scheduling in sensor networks: Distributed edge coloring revisited. *J. Paral. Distrib. Comput.* **2008**, *68*, 8.
33. Ma, J.; Lou, W.; Wu, Y.; Li, X.-Y. Energy efficient TDMA sleep scheduling in wireless sensor networks. In Proceedings of INFOCOM 2009, Rio de Janeiro, Brazil, April 2009.
34. Rajendran, V.; Obraczka, K.; Garcia-Luna-Aceves, J.-J. Energy-efficient, collision-free medium access control for wireless sensor networks. In Proceedings of Sensys'03, Los Angeles, CA, USA, November 2003.
35. Rajendran, V.; Garcia-Luna-Aceves, J.J.; Obraczka, K. Energy-efficient, application-aware medium access for sensor networks. In Proceedings of IEEE MASS 2005, Washington, DC, USA, November 2005.
36. Gobriel, S.; Mosse, D.; Cleric, R. TDMA-ASAP: Sensor network TDMA scheduling with adaptive slot stealing and parallelism. In Proceedings of ICDCS 2009, Montreal, Canada, June 2009.
37. Rhee, I.; Warrier, A.; Aia, M.; Min, J. Z-MAC: A hybrid MAC for wireless sensor networks. In Proceedings of SenSys'05, San Diego, CA, USA, November 2005.
38. Rhee, I.; Warrier, A.; Xu, L. *Randomized Dining Philosophers to TDMA Scheduling in Wireless Sensor Networks*; Technical Report TR-2005-21; North Carolina State University: Raleigh, NC, USA, April 2005.
39. Garey, M.; Johnson, D. *Computers and Intractability: A Guide to the Theory of NP-completeness*; Freeman: San Francisco, CA, USA, 1979.
40. Brelaz, D. New methods to color the vertices of a graph. *Commun. ACM* **1979**, *22*, 251–256.
41. Peemoller, J. A correction to Brelaz's modification of Brown's coloring algorithm. *Commun. ACM* **1983**, *26*, 595–597.
42. Johansson, O. Simple distributed $(\Delta + 1)$ -coloring of graphs. *Inform. Process. Lett.* **1999**, *70*, 229–232.
43. Busch, C.; Magdon-Ismail, M.; Sivrikaya, F.; Yener, B. Contention-free MAC protocols for wireless sensor networks. In Proceedings of DISC 2004, Amsterdam, The Netherlands, October 2004.
44. Finoccho, I.; Panconesi, A.; Silvestri, R.; Experimental analysis of simple distributed vertex coloring algorithms. In Proceedings of SODA 2002, San Francisco, CA, USA, January 2002.
45. Kuhn, F.; Wattenhofer, R. On the complexity of distributed graph coloring. In Proceedings of PODC 2006, Denver, CO, USA, July 2006.
46. Hansen, J.; Kubale, M.; Kuszner, L.; Nadolski, A. Distributed largest-first algorithm for graph coloring. *Lect. Note. Comput. Sci.* **2004**, *3149*, 804–811.
47. Durand, D.; Jain, R.; Tseytlin, D. Distributed Scheduling Algorithms to Improve the Performance of Parallel Data Transfers. In Proceedings of IPPS'94 Workshop on Input/Output in Parallel Computer Systems, Cancun, Mexico, April 1994.
48. Tamura, H.; Watanabe, K.; Sengoku, M.; Shinoda, S. On a New Edge Coloring Related to Multi-hop Wireless Networks. In Proceedings of APCCAS'02, Singapore, October 2002.

49. Barrett, T.; Istrate, G.; Kumar, A.; Marathe, M.; Thite, S.; Thulasidasan, S. Strong edge coloring for channel assignment in wireless radio networks. In Proceedings of IEEE International Workshop on Foundation and Algorithms for Wireless Networking, Pisa, Italy, March 2006.
50. Kunvilaisah, W.; Hou, Y.; Zhang, Q.; Zhu, W.; Kuo, C.; Zhang, Y. A cross layer quality of service mapping architecture for video delivery in wireless networks, *IEEE J. Sel. Area. Commun.* **2003**, *21*, 10.
51. Kawadia, V.; Kumar, P. A cautionary perspective in cross layer design. *IEEE Wirel. Commun.* **2005**, *12*, 3–11.
52. Schaar, M.; Shankar, S. Cross layer wireless multimedia transmission: Challenges, principles and new paradigms. *IEEE Wirel. Commun.* **2005**, *12*, 50–58.
53. Winter, R.; Schiller, J.; Nikaem, N.; Bonnet, C. Crosstalk: Cross layer decision support based on global knowledge. *IEEE Commun. Mag.* **2006**, *44*, 2–8.
54. Raisinghani, V.; Iyer, S. ECLAIR: An efficient cross layer architecture for wireless protocol stacks. In Proceedings of 5th World Wireless Congress, San Francisco, CA, USA, May 2004.
55. Zhu, J.; Chen, S.; Bensaou, S.; Hung, K. Tradeoff between Lifetime and Rate Allocation in Wireless Sensor Networks: A Cross Layer Approach. In Proceedings of IEEE INFOCOM 2007, Anchorage, AK, USA, May 2007.
56. Nama, H.; Chiang, M.; Mandayam, N. Optimal Utility-lifetime Trade-off in Self-regulating Wireless Sensor Networks. A Distributed Approach. In Proceedings of CISS 2006: IEEE Conference on Information, Sciences and Systems, Princeton, NJ, USA, March 2006.
57. Nama, H.; Chiang, M.; Mandayam, N. Utility-lifetime Trade-off in Self-regulating Wireless Sensor Networks. A Cross-layer Design Approach. In Proceedings of IEEE ICC 2006, Istanbul, Turkey, May 2006.
58. Kwon, H.; Kim, T.; Choi, S.; Lee, B.G. A Cross-layer Strategy for Energy-efficient Reliable Delivery in Wireless Sensor Networks. *IEEE Trans. Wirel. Commun.* **2006**, *5*, 12.
59. Adjih, C.; Clausen, T.; Jacquet, P.; Laouiti, A.; Minet, P.; Muhlethaler, P.; Qayyum, A.; Viennot, L. *Optimized Link State Routing Protocol*; RFC 3626; The Internet Society: Reston, VA, USA, 2003.
60. Macker, J.; Downard, I.; Dean, J.; Adamson, B. Evaluation of Distributed Cover Set Algorithms in Mobile *Ad Hoc* Network For Simplified Multicast Forwarding. In Proceedings of ACM SIGMOBILE Mobile Computing and Communications Review, New York, NY, USA, July 2007.
61. Mao, J.; Wu, Y.; Li, X.-Y.; Chen, G. Energy efficient TDMA sleep scheduling in wireless sensor networks. In Proceedings of INFOCOM 2009, Rio de Janeiro, Brazil, April 2009.