



HAL
open science

RTXP : A Localized Real-Time Mac-Routing Protocol for Wireless Sensor Networks

Alexandre Mouradian, Isabelle Augé-Blum, Fabrice Valois

► **To cite this version:**

Alexandre Mouradian, Isabelle Augé-Blum, Fabrice Valois. RTXP : A Localized Real-Time Mac-Routing Protocol for Wireless Sensor Networks. [Research Report] RR-7978, 2012. hal-00702302v1

HAL Id: hal-00702302

<https://inria.hal.science/hal-00702302v1>

Submitted on 29 May 2012 (v1), last revised 13 Mar 2013 (v2)

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

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



RTXP : A Localized Real-Time Mac-Routing Protocol for Wireless Sensor Networks

Alexandre Mouradian, Isabelle Augé-Blum , Fabrice Valois

**RESEARCH
REPORT**

N° 7978

May 2012

Project-Team urbanet



RTXP : A Localized Real-Time Mac-Routing Protocol for Wireless Sensor Networks

Alexandre Mouradian*, Isabelle Augé-Blum*, Fabrice Valois*

Project-Team urbanet

Research Report n° 7978 — May 2012 — 26 pages

Abstract: Protocols developed during the last years for Wireless Sensor Networks (WSNs) are mainly focused on energy-consumption optimization and autonomous mechanisms (e.g. self-organization, self-configuration, etc). Nevertheless with new WSN applications appear new QoS requirements such as time constraints. Real-time applications require the packets to be delivered before a known time bound which depends on the application requirements. We particularly focus on applications which consist in alarms that are sent to the sink node. We propose Real-Time X-layer Protocol (RTXP) a real-time communication protocol. This protocol integrates both MAC and routing layers. Our proposition aims at guaranteeing an end-to-end constraint delay while keeping good performances on other parameters, such as energy consumption. For this purpose the protocol relies on a hop-count-based Virtual Coordinate System (VCS) which classifies nodes having the same hop-count from the sink, allows forwarder selection, and gives to the nodes an unique identifier in a 2-hop neighborhood allowing deterministic medium access. In this document we describe the protocol mechanisms and properties. A theoretical comparison of our proposition with a state-of-the-art protocol is given. Intensive simulation results confirm the theoretical predictions. We also discuss the performances of the protocol under the assumption of unreliable radio links.

Key-words: wireless sensor networks, virtual coordinate system, application constraints, MAC, routing

* University of Lyon, INRIA, France, INSA Lyon, CITI, F-69621, France

**RESEARCH CENTRE
GRENOBLE – RHÔNE-ALPES**

Inovallée
655 avenue de l'Europe Montbonnot
38334 Saint Ismier Cedex

RTXP : A Localized Real-Time Mac-Routing Protocol for Wireless Sensor Networks

Résumé : Les protocoles développés pour les réseaux de capteurs sans fil sont conçus principalement pour réduire la consommation d'énergie et permettre au réseau de s'auto-organiser. Les applications critiques temps-réel nécessitent, en plus de ces caractéristiques, que les communications soient fiables et que le délai de bout en bout soit borné. Dans ce document nous nous intéressons principalement à des remontées d'alarmes vers le puits. Nous proposons RTXP (Real-Time X-layer Protocol) un protocole de communication temps-réel. Ce protocole définit les mécanismes des niveaux MAC et routage. Le but est de garantir une contrainte sur le délai de bout en bout tout en conservant des bonnes performances pour les autres paramètres tel que la consommation d'énergie. Pour cela le protocole utilise un système de coordonnées virtuelles qui classe les noeuds qui ont le même nombre de sauts depuis le puits. Cela sert à choisir le noeud qui doit relayer un paquet. Ce système de coordonnées permet aussi de différencier les noeuds d'un deux-voisinage ce qui autorise à avoir des accès au médium déterministes. Dans ce document nous décrivons les mécanismes de ce protocole, nous le comparons à un protocole existant. Une campagne de simulation permet de confirmer les résultats théoriques. Nous discutons les performances de ce protocole dans le cas où les liens radio sont non fiables.

Mots-clés : réseaux de capteurs sans fil, système de coordonnées virtuelles, contraintes applicatives, MAC, routage

1 Introduction

A WSN is composed of nodes deployed in an area in order to monitor parameters of the environment. Those nodes are able to send information to dedicated nodes called sinks without the need of a fixed network infrastructure. Every node is able to forward messages from the other nodes. They usually run on batteries so they should consume as little energy as possible in order to increase the network lifetime. Because WSNs can contain thousands of nodes, the cost of a node should be as low as possible, this leads to design nodes with poor capabilities (computation, radio, memory, etc...). In the past few years WSNs have been a very active research field which has led to interesting contributions at all communication layers. This is due to the great expectations put in WSN applications. In fact, lots of applications have been proposed in the literature, such as water-metering [12], volcano monitoring [27], air pollution monitoring [17], landslide detection [21] and so on.

Due to the previously mentioned characteristics of WSNs, network protocols have been designed mainly in order to optimize energy consumption and to provide autonomous network mechanisms. Nevertheless some applications need more than these characteristics. Indeed, critical applications require more reliability and the respect of time constraints. For instance the aforementioned landslide detection application should give guarantees on the delivery of alert messages. Protocols which can deliver messages with guaranteed end-to-end delay are called real-time protocols. They are usually classified into two categories, soft real-time and hard real-time: in the first case, the delay constraint may not be respected in some cases while in the second case, the delay constraint should be always respected whatever the circumstances are because of the possible impact on human life or on the financial cost. Due to the probabilistic nature of the radio links in WSNs, the hard real-time protocols must provide reliability. This parameter is thus a main concern in the design of a WSN real-time protocol.

Hard real-time constraints cannot be met with the current WSN protocols of the literature either because of their lack of determinism which implies unbounded delays or low reliability, or because they do not take into account the aforementioned characteristics of WSNs.

In order to handle such requirements, deterministic mechanisms must be introduced at MAC and routing layer. The interactions between these two layers must also be carefully controlled in order to avoid unexpected and unbounded delays. We thus claim that a cross-layer design where MAC and routing layers share information is required.

In this document we propose RTXP, a real-time cross-layer (X-layer) protocol. This protocol aims at giving a bound on the end-to-end delay in a WSN. Our approach is to bound the duration of one hop and the number of hops to reach the sink. The access to the medium and the choice of the forwarder must be deterministic. Our approach is based on a suitable VCS[19]. This VCS allows the nodes to get information on their distance to the sink in number of hops. It also differentiates nodes having the same hop-counts in order to select forwarders. Finally, It gives an unique identifier to the nodes in a 2-hop neighborhood in order to deterministically access the medium.

In section 2 we discuss the advantages and drawbacks of existing MAC and routing protocols for WSN. In section 3 we introduce the hypothesis and the

requirements of our solution. In section 4 we propose RTXP and discuss its properties. In section 5 we describe a state-of-the-art protocol and compare it with our proposition. In section 6 we present the simulation parameters and results and discuss the performances of RTXP. In section 7 we conclude on the protocol properties and performances and we present our future works.

2 Related work

WSN MAC and routing protocols were widely investigated during the last years. Unfortunately only few contributions were focused on real-time protocols. In this section we discuss the main results for MAC only, routing only and cross-layer protocols.

2.1 Medium Access Control

MAC protocols can be classified into two main categories: synchronous and asynchronous. In synchronous protocols, the nodes know the schedules of the actions of other nodes (in their neighborhood or in the whole network). Usually a mechanism is used to synchronize the clocks of the nodes. They thus share a common global or local clock. In asynchronous protocols, the synchronization exists but it is event-based. A communicating node and its neighborhood synchronize only for the time of a communication but without exchanging the values of their clocks. Usually a duty cycle [20] mechanism is used to save energy. Since the receiving, sending and listening energy costs are approximately the same for usual radio chips [1], the only way to save energy is to turn off the radio (to switch to sleep mode). Thus duty-cycling consists in nodes alternately waking up and going to sleep mode. With synchronous protocols the wakeup times of the nodes are known by the others (at least in a 1 hop neighborhood). With asynchronous protocols the nodes do not know the wakeup times of their neighbors. In both cases, the medium can be accessed either by contention or in a deterministic way.

BMAC [20] illustrates a classical asynchronous protocol for WSNs. First, nodes pick a random wakeup time and then alternately sleep and wakeup. BMAC uses LPL (Low Power Listening) method which consists in listening periodically to the channel for radio activity. When a node needs to send a message, it sends a preamble (sequence of bits) which duration is equal to the duty-cycle period. Each node, when it wakes up, senses the channel, if it detects the preamble it stays awake until the end of the communication in order to receive the packet, otherwise it goes back to sleep. The main advantage of this protocol is that there is no need for a time synchronization algorithm. Nevertheless the length of the preamble and the overhearing problem (nodes listening the preamble even if they are not the destination of the packet) can lead to high energy consumption. Enhancements have been proposed by XMAC [5] and MFP [4] for instance. They aim at reducing energy consumption by avoiding unnecessary listening.

SMAC [30] is a synchronous protocol. All the nodes in a neighborhood synchronize in order to wake up at the same time and access the medium by contention using carrier sense and RTS/CTS mechanisms. The protocol is localized, thus scalable. D-MAC [18] enhances this mechanism by scheduling the

wakeup time in function of the distance to the sink. This reduces the end-to-end delay of the packets in the network.

In the previously cited protocols the channel access is not deterministic (i.e. collisions can occur). It implies that the time to access the medium is not guaranteed. This leads to unbounded delay to perform one hop which is not suitable for real-time applications.

Solutions that provide deterministic access to the medium have been proposed, such solutions allow to respect real-time constraints.

I-EDF [6] proposes an approach based on Early Deadline First scheduling algorithm. I-EDF is a synchronous protocol. The network is divided in hexagonal cells which use different radio frequencies. There are two kinds of communications, the intra-cell communications and the inter-cell communications. For intra-cell communications, all the nodes of a cell know the periodicity and length of the packets of every other node, in the same cell, so they can apply the EDF algorithm. Between two intra-cell communications there are slots reserved for inter-cell communications. There are 6 directions in which a node can emit (corresponding to the six edges of the hexagonal cell) each inter-cell slot is allocated to a direction (each router emitting on a different frequency which corresponds to the one of its neighbor in the current direction). This proposition allows real-time communications but does not take into account the energy consumption. Moreover the nodes must have the capacity of communicating on 6 channels and the hexagonal cell topology is restraining.

PR-MAC [8] is also a synchronous real-time MAC for WSNs. The aim is to detect events and then to set up a periodical monitoring of the area where the event occurred. When an event is sensed in the network an alarm is sent to the sink. The sink responds with a packet that reserves a path for the periodical monitoring. The nodes on the path then wake up two times, once for the traffic from the source to the sink and once for the traffic from the sink to the source. The path is reserved with a given radio frequency. The construction of the path depends on the routing protocol used. Once the path is reserved the monitoring packets are transmitted in real-time but the reservation phase is non real-time and induces an overhead. Moreover the protocol assumes the radio handles multi-channel communications.

On the contrary f-MAC [23] proposes a localized and asynchronous approach. The principle is that nodes periodically send small packets (called frames) with a dedicated period, each node in the neighborhood having a unique period attributed. The authors show that, by applying mathematic rules for the choice of the periods, it can be guaranteed that a frame of each node will actually be transmitted without collision. This MAC guarantees hard real-time constraints on perfect radio links and the transmission mechanisms are very simple. Nevertheless it has very a poor channel utilization, a quite high energy consumption and the maximum delay increases exponentially with the number of nodes in the same collision domain.

Dual-MAC [29] is a hard real-time MAC for linear WSNs. Thanks to the linear topology of the WSN, the routing problem has not to be taken into account. At initialization the network is divided into cells. The protocol can operate in two modes. The “unprotected mode” allows a near-optimal transmission time when there is no collision. When a collision occurs, a “protected mode” is triggered in which a path is reserved in order to guarantee bounded transmission times. The furthest node from the sender is elected as the packet relay. The

two modes allow the protocol to provide a good trade-off between average performances and respect of strict time constraints. The worst case is computed and the protocol is formally verified with a model checker. Nevertheless the solution is only suited for linear networks and does not take into account energy consumption issues in the design of the protocol.

The MAC protocols described in this document do not allow to respect strict time constraints (hard real-time) while taking into account the previously cited requirements specific to WSNs. The propositions that allows to respect hard time constraints do not take into account energy consumptions issues, radio chip limitation or are difficult to integrate with a routing layer. Other propositions are more suited to WSNs but do not allow to respect strict time constraints.

2.2 Routing

In this section we focus on routing protocols for WSNs. They can be classified into four categories: probabilistic, hierarchical, location-based and broadcast-based.

In probabilistic routing protocols, such as Random Walk Routing [25], forwarders are elected by making random choices. This class of protocol cannot be used for real-time communications because of its lack of determinism. Indeed, it leads to unbounded routes length which do not allow to provide a bound on end-to-end delay.

In hierarchical protocols nodes are grouped into clusters. A leader called cluster head is a node which is responsible for collecting the data sensed by the nodes of the cluster and sending it to the sink. In this case the length of the route is bounded. Nevertheless, maintaining the hierarchical structure can be expensive in terms of energy consumption in highly dynamic networks. Moreover, in the case of a cluster head failure many nodes are disconnected from the sink.

Location-based protocols are making forwarding decisions depending on the geographic location of the destination of the packet. A method for choosing a forwarder is to elect the neighbor of the sender which is the closest in distance to the sink. This mechanism is called greedy forwarding and has good performances in WSNs [16]. The distance can be geographic distance [16] but [28] shows that lack of accuracy of the geographic coordinates leads to bad performances. Furthermore GPS chips cost is high and the energy consumption is not negligible. VCSs have been developed in order to address those issues. [22] and [7] propose two different types of VCS, greedy routing is then used with the coordinates. With such a technique the number of hops is not known a priori. Nevertheless a bound on the number of hops could be deduced from the position of the furthest node from the sink. This solution is however only applicable in dense networks. Indeed, a hole in the network can induce a longer route than expected [16]. This can be an issue in the case of real-time communications because we cannot bound the hop-count of a packet and thus the time it takes to reach the sink. SPEED [26] is a soft real-time protocol based on geographic coordinates. A node keeps a table of its neighbors with a metric that represents their speed. The speed of a neighbor is computed by dividing the advance in geographic distance it provides in direction of the destination, by the delay to forward the packet to that neighbor. The forwarder is selected if its speed allows to respect the deadline. SPEED does not prevent contention thus

it is soft real-time. Nevertheless, it provides a congestion detection mechanisms. MMSPEED [14] enhances reliability of SPEED by using a multi-path scheme.

In broadcast-based routing protocols, a node does not need to store explicit information on the network topology. It broadcasts the message and the choice of the next hop is done by nodes which receive it. The choice is based on a metric that can depend on the coordinates (geographic or virtual) and other parameters of the potential forwarders or it can be done randomly. GRAB [30] can be classified in this category of routing protocols. GRAB introduces a cost-field which can be seen as a VCS or a metric, indeed the cost-field represents the cost for a node to reach the sink. In GRAB the hop-count is used as a cost-field. During the initialization phase each node is assigned its distance to the sink, in number of hops, as coordinate. Then packets are routed using gradient-routing which consists in choosing the forwarder which has the lowest cost-field value. As many nodes with the same hop-count can listen the packet, the selection of the forwarder can be based on a random value and multiple forwarders can be elected, this induces multiple paths. The advantages of such a solution are that the number of hops to reach the sink is known and multiple path leads to more reliability. Nevertheless GRAB does not give information on the physical organization of nodes with the same hop-count. This information could be useful in order for example to select the best forwarder for a packet in a deterministic way (in the case of greedy routing it is the nearest from the sink), it could be used as well to access the channel in a X-layer scheme. SGF [15] and LQER [9] propose similar schemes. In SGF only one node is chosen in an opportunistic manner. LQER add information on the link quality. Both solutions suffer from the same aforementioned drawbacks of GRAB.

2.3 Cross-layer

Solutions which integrate both MAC and routing mechanisms have been proposed. These solutions allow to plan routes and medium access simultaneously. TSMP [11] uses a MF-TDMA scheme to access the medium. It uses a centralized scheduling, where time-slots and channels are assigned to nodes in order to avoid interferences.

PEDAMACS [13] also uses such a scheme but with only one radio channel. Nodes have different transmission ranges. The sink can reach all the nodes in the network, the other nodes have two transmission ranges: one to communicate and one to identify their interferers. The protocol needs a global synchronization of the network. This is achieved thanks to synchronization packets that are sent by the sink to the whole network. The protocol consists in three phases, in the first one, the topology learning phase, each node learns its interferers and neighbors by sending hello packets in contention periods. During the second phase, the topology collection phase, the information is sent to the sink using a contention mechanism. A schedule is produced by the sink and sent to the nodes. The method used to produce the schedule is to linearize the graph of the network (containing the interference edges) and to give the same color to non interfering levels. The slots are allocated to non-interfering sets of nodes with the same color. During the third phase the nodes communicate in their allocated slots.

A drawback of centralized protocols is that the sink needs to retrieve information on the topology which is not scalable and can lead to high energy

consumption and memory issues. Also, the sink is a weak point of the system: if it fails everything fails.

Further in this document we study the differences between PEDAMACS and our proposition mechanisms and the trade-offs those differences underline. We choose PEDAMACS because to the best of our knowledge it is the only protocol that integrates MAC and routing layers being able to guarantee real-time delivery of packets with the hypothesis that there is only one channel available and taking into account energy issues. Moreover, it gives us the opportunity to compare centralized and localized real-time protocols.

In the previous paragraphs we present different solutions to schedule the access to the channel and route packets in WSNs. Nevertheless, none defines both MAC and routing layers that can ensure bounded end-to-end delays while being localized, thus scalable. Our solution is based on a deterministic and localized access scheme and on greedy and opportunistic routing mechanisms. The coordinate is used at MAC layer in order to bound access to the medium and routing layer to choose an unique forwarder.

3 Hypothesis and problem statement

3.1 Hypothesis

In this section we discuss the assumptions we consider for the remainder of the document. There are several kinds of assumptions, some induced by the limited capacities of sensor nodes, some linked to the radio, some linked to the applications and others more specific to our proposition.

Assumptions on the limited capacities of sensor nodes:

- Sensors have a limited amount of energy;
- The nodes have a limited amount of memory.

Assumptions linked to the radio:

- The radio is half-duplex and mono-channel;
- Nodes can decode packets from their 1 hop neighbors and listen radio activity from their 2 hop neighbors;
- For the design of our solution we assume that the radio link is perfect i.e., packet loss is only due to collision. This hypothesis is then relaxed in the performances evaluation part (see section 6).
- Radio links are symmetric.

Assumptions linked to the application:

- The traffic is low and consists in alarm packets converging toward the sink;
- The sensor nodes clocks have to be synchronized [3] so they can wake up at the same time.

Assumptions more specific to our propositions:

- A local data aggregation mechanism has been performed when the event occurs, before the alarm is sent, in order to avoid the transmission of redundant information;
- Nodes have local coordinates which give information on the number of hops from the sink and which are unique in a 2 hops neighborhood. This provided by the mechanisms of [19].

3.2 Problem statement

We want to find a protocol in order to establish real-time communications in WSNs. The characteristics of WSNs impose to this protocol to respect requirements. It should be scalable due to the large scale of WSNs, it should be reliable because the applications are critical and the wireless links are unreliable. The protocol should ensure that the end-to-end delay is lower than a given bound. In the remainder of the document we propose and evaluate RTXP, a protocol which respect those requirements.

4 Proposition: a novel Real-Time X-layer Protocol, RTXP

In this section we present RTXP, a cross-layer (MAC and routing) protocol which guarantees a bounded end-to-end delay for packets (under reliable links hypothesis). We first give the general ideas of the protocol, we describe the virtual coordinate it uses as a metric, we go further into the mechanisms of the protocol, we compute theoretical parameters of the protocol and finally we discuss trade-offs on these parameters.

4.1 General idea

As energy is a main concern in WSNs, RTXP uses a duty-cycle mechanism. We call *activity period* the period in which the nodes are awake and *sleep period* the one in which they turn off their radio.

The activity period is divided into three main phases, a contention period to access the channel, the transmission of the packet and a contention period to elect the forwarder. The decision to be a forwarder is local and opportunistic. Three strong principles are:

1. the medium accesses and forwarder selections are deterministic;
2. if a node has a message at the beginning of the duty cycle period it will be forwarded to a node with a lower hop-count by the end of the period;
3. nodes which are 3 hop-count distant can communicate simultaneously since they do not interfere with each others (a node with hop-count n interferes with rings $n - 2$, $n - 1$, $n + 1$ and $n + 2$).

As stated in section 3 we assume that all the nodes are synchronized [3]. We also assume that coordinate initialization has been performed so the coordinates are unique in every 2 hops neighborhoods and the nodes with the same hop-count are differentiated. We describe the coordinate system we use in the next section.

4.2 Virtual coordinate system

The VCS used by RTXP consists in a 1-D coordinate, which has been proposed in [19], and is based on two parameters. The first one is the hop-count to the destination node, but many nodes can have the same hop-count. The nodes having the same hop-count can be seen, conceptually, as forming concentric rings centered on the sink. So the second information is the connectivity of a node with the different rings. These two parameters are then integrated into one coordinate which classifies the nodes of a same ring. This classification is related to the connectivity, nodes having more neighbors in proportion in the rings nearer to the sink are classified before nodes having more neighbors in proportion in the rings further from the sink. This information allows to give priority to nodes more connected to lower rings for the forwarders selection. Figure 1 illustrates the coordinate, R is the radio range of a node, $offset$ embed the information on connectivity (c.f. [19] for more details) and n is a ring number. The coordinate is given by $Coord = (n-1)*R + offset$ with $offset < R$. In the proposed solution the probability of having two nodes with the same coordinate in a 2-hop neighborhood is low but not null (this issue is discussed in [19]). In this document we assume that the coordinate is unique in a 2-hop neighborhood.

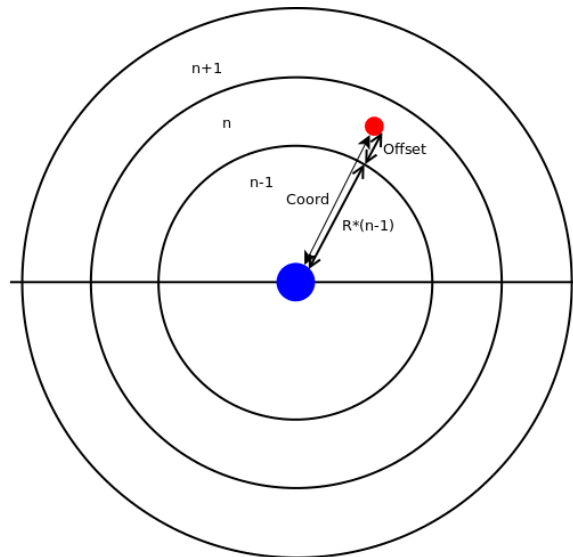


Figure 1: Illustration of the 1-D coordinate used by RTXP

4.3 In-depth detail of RTXP

The three phases of the protocol are depicted on Figure 2. The notations used in this section are detailed in table 1.

1. In the first phase, the contention period, each node being awake and having a packet to send contends for the channel. During the contention a node senses the channel. If it detects energy on the channel before the end of its backoff timer it loses the contention otherwise it sends a jamming code. A

jamming code is a short sequence of bits, possibly random. The technique is very similar to preamble sampling [20] but with the jamming code being shorter than a typical preamble. If a node loses the contention it can notify it in a dedicated slot (noted L for Lost on Figure 2) at the end of the activity period by sending a jamming code in it. Every node that receives a jamming code in the L slot will stay awake for another activity period, if the end of the sleep period is not reached. The nodes are able to hear jamming codes from their 2-hop neighborhood in order to prevent the hidden terminal problem. The backoff timer is calculated with a bijective function from the coordinate. The lemma 4.1 means that there is no collision in a two-hop neighborhood

Lemma 4.1 *If the coordinates are unique in a 2-hop neighborhood and the back-off function is bijective then there is only one node that wins the contention.*

Proof We do a proof by contradiction. Let's suppose there are two nodes that win the contention (i.e. there is a collision). That means they have the same backoff time which implies that either they have the same coordinate or the backoff function is not bijective which is a contradiction.

B : Backoff phase
 BF : Backoff Forward phase
 R : Receive phase for ring n (corresponding to sending phase for ring n+1)
 L : Slot used to notify the loose of a contention
 n : Hop-count to reach the sink

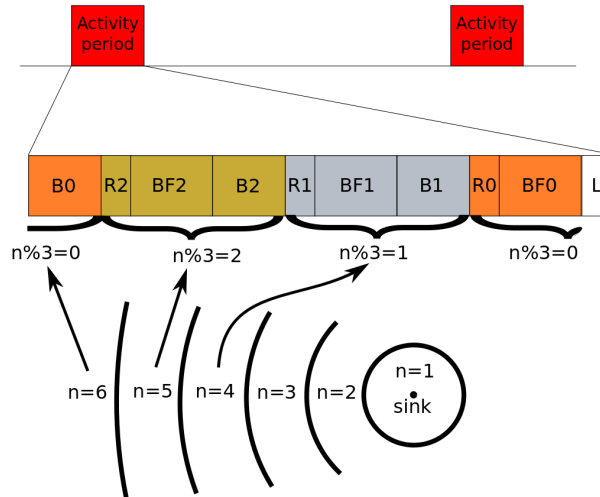


Figure 2: Description of the proposition

2. During the second phase, the node who won the contention sends its packet and the nodes at ring n-1 receive it.

3. The third phase is another contention period. All the nodes that received a packet in the second phase contend to know which one will forward it. As in the first phase, the backoff function is bijective and calculated from the coordinate. The first node which backoff ends sends a jamming code to notify the others

that it will be the forwarder. We can notice that this mechanism is a kind of opportunistic forwarding based on the coordinate.

Figure 2 depicts the different phases for nodes in different rings. B_i , R_i and BF_i correspond respectively to backoff, receive and backoff forward phases with $i = n \bmod 3$. For example a node in the ring 6 (6 hops from the sink) contends in B_0 if it has a packet to send. It sends the packet in R_2 if it wins the contention. It wakes up in R_0 to potentially receive a packet and if it has received one it does the BF_0 phase to try to forward the packet. The L slot corresponds to the slot where nodes that have lost contention can notify it. It keeps potential forwarders awoken at the end of the activity period.

symbol	signification
D_B	Duration of the B phase
D_{BF}	Duration of the BF phase
D_R	Duration of the R phase
D_L	Duration of the L slot
$D_{jamming}$	Duration of the jamming code
$D_{activity_period}$	Duration of an activity period
D_{max_awake}	The maximum amount of time a node can be awake during an activity period
D_{sleep}	Duration of the sleep period
$WCTT_{RTXP}$	Theoretical bound on the end-to-end delay for RTXP
DC	The duty-cycle ratio
C_{RTXP}	The capacity of RTXP
N_{Bhop_max}	The maximum number of hops from the sink
$E_{backoff}$	Energy consumed during the B phase
$E_{backoff_forward}$	Energy consumed during the BF phase
$E_{snd_jamming}$	Energy consumed during the emission of a jamming code
E_{snd_packet}	Energy consumed during the emission of a packet
E_{rcv_packet}	Energy consumed during the reception of a packet
E_{1hop_RTXP}	Energy consumed by RTXP to do one hop
$E_{initialization_1node}$	Energy needed to initialize one node
$E_{snd_hello_packet}$	Energy consumed during the emission of a hello packet
$E_{rcv_hello_packet}$	Energy consumed during the reception of a hello packet
k	Number of neighbors of a node

Table 1: Notations used in the description of RTXP

Dimensioning B and BF phases must be done carefully in order to guarantee that there is actually no collision. This highly depends on the backoff which is function of the offset ($backoff = BO(offset)$). Indeed, each node must have the possibility to send a jamming code during the period so the duration of the backoff phases must be greater than the maximum backoff duration plus the duration of a jamming code:

$$D_B = D_{BF} \geq \max(BO) + D_{jamming}$$

Moreover the radio must be able to sample the channel, with a period that is smaller than the difference between any two backoff durations. If it is not the case, a collision can occur because a node may not sense the beginning of a jamming code before it emits its own jamming code.

The R phase duration is the time needed to transmit a data packet (noted D_R). In our case the data packet is an alarm packet which size is in the order of magnitude of a few dozens of bytes.

The dimensioning of these periods highly depends on the characteristics of the radio chip used for the deployment, the higher the bitrate, the shorter the periods. The sleep period duration (D_{sleep}) is calculated based on the maximum duration a node is awake during the activity period (D_{max_awake}). This maximum duration corresponds to the B phase plus the BF phase plus two R phases (one to send a message and one to receive one). The calculation of D_{sleep} depends on the duty-cycle ratio wanted, $DC = D_{max_awake}/(D_{sleep} + D_{max_awake})$:

$$D_{max_awake} = D_B + D_{BF} + 2 \times D_R$$

$$D_{sleep} = D_{max_awake} \times \left(\frac{1}{DC} - 1 \right)$$

The duty-cycle ratio typically depends on the application characteristics, this aspect is discussed in section 4.5.

The activity period is represented on the figure 2 its duration is given by:

$$D_{activity_period} = 3 \times (D_B + D_{BF} + D_R) + D_L$$

A node which loses the contention in B phase can retry to contend in another activity period. Indeed, if a node receives a jamming code in the L slot, it stays awake for a new activity period.

The figure 3 shows an example with a simple network. The nodes A and B have both data to send to the sink at the beginning. They contend in the first part of the activity period. B wins the contention so it can send its packet in the R phase to C and D, then forward it until it reaches the sink. At the end of the first activity period, node A sends a jamming code in the L slot because it just lost the contention previously. B, C and D listen the jamming code so they stay awake for a new activity period. In this second activity period B has no packet to send anymore thus A wins the contention and the packet is forwarded to C.

4.4 Capacity, delay and energy

The number of activity periods is limited by the length of the sleep period (we can notice that it depends on the duty-cycle value). This limit corresponds to the capacity C_{RTXP} of our protocol, beyond this capacity real-time communication are not ensured. The capacity is given in number of packets that can be transmitted in a 2-hop neighborhood during ($D_{activity_period} + D_{sleep}$) and is given by:

$$C_{RTXP} = \frac{D_{activity_period} + D_{sleep}}{D_{activity_period}}$$

Theorem 4.2 *Let $n \in \mathbb{N}$ and $p \in \mathbb{N}$ be respectively a ring number and the number of packets in a 2-hop neighborhood at ring n . All packets in every 2-hop neighborhood at ring n will reach ring $n - 1$ in at most a duty cycle period if $p < C_{RTXP}$.*

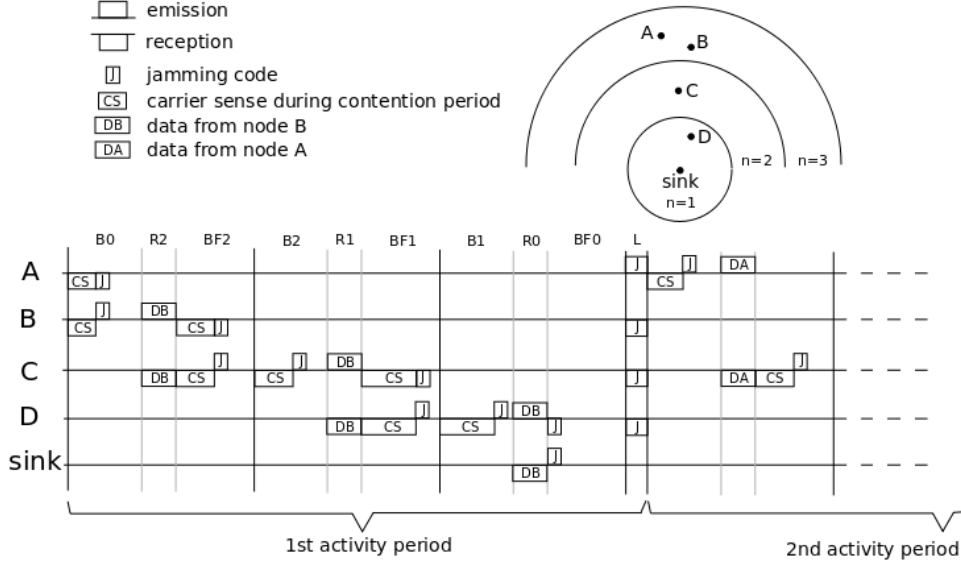


Figure 3: Example with 4 nodes where nodes A and B have a packet to transmit

Proof We do a proof by contradiction. Let's suppose one packet did not reach ring $n - 1$ in one duty-cycle period. Then either the packet was lost or it was delayed until the end of the period. As we assumed earlier the only way to lose a packet is because of a collision. By lemma 4.1 we know that it is not possible so it is a contradiction. If the packet is delayed until the end of the duty-cycle period that means the node lost every contention until the end for that packet. So it lost more than $\frac{D_{activity_period} + D_{sleep}}{D_{activity_period}} = C_{RTXP}$ contentions, so there were more than C_{RTXP} packets ($p > C_{RTXP}$) which is a contradiction.

Thus under this capacity limit the delivery ratio is 100%. We can notice that a packet can do at most three hops during one activity period if the nodes, which forward it, do not lose any contention. If one node loses a contention, the packet may do only one hop because the node has to notify the loss in the L slot, and then, the $n - 2$ and $n - 3$ rings may not be awake anymore.

In order to compute a bound on the end-to-end delay, we propose to calculate the Worst Case Transmission Time (WCTT) for one hop and multiply it by the maximum number of hops in the network. The WCTT for one hop is given by $D_{activity_period} + D_{sleep}$ because a packet, in the worst case, is transmitted after having lost the contention every time until the limit given by the end of the sleep period. We take the maximum number of hops plus one because there is a delay (which is at most one duty cycle period) between the instant the event is sensed and the first emission of the corresponding packet.

$$WCTT_{RTXP} = (NB_{hop_max} + 1) \times (D_{activity_period} + D_{sleep})$$

The energy used during one hop if the relaying node wins the contention is:

$$\begin{aligned} E_{1hop_RTXP} = & E_{backoff} + E_{snd_jamming} + E_{snd_packet} \\ & + E_{rcv_packet} + E_{backoff_forward} \\ & + E_{snd_jamming} \end{aligned}$$

From [19] we can deduce the amount of energy $E_{initialization_1node}$ needed by a node in order to initialize its coordinate.

$$E_{initialization_1node} = E_{snd_hello_packet} + k \times E_{rcv_hello_packet}$$

4.5 Trade-offs

The capacity depends on the inverse of the duty cycle ratio, so the longer the sleep period the higher the capacity. Moreover with a longer sleep period more energy is saved. Nevertheless the bound on the delay of a packet increases with the sleep period. Thus a trade-off which depends on the application has to be found between the bound on the delay and the capacity of the protocol. In the case of applications with low traffic and short time constraints a small sleep period should be used. In the case of applications with high traffic and less tight time constraints a longer sleep period should be preferred.

5 Theoretical study: comparison between RTXP and PEDAMACS mechanisms

Our proposition is, to the best of our knowledge, the first hard real-time localized cross-layer protocol for WSNs. It is thus difficult to fairly compare its performances with an existing solution. Nevertheless, we choose to compare the mechanisms of RTXP with those of PEDAMACS [13] in order to underline the trade-offs in the design of our proposition. We choose PEDAMACS because, to the best of our knowledge, it is the only protocol that integrates MAC and routing layers being able to guarantee real-time delivery of packets with the hypothesis that there is only one channel available and taking into account energy issues. PEDAMACS protocol is centralized, this characteristic makes it difficult to fairly compare the two protocols. Nevertheless the comparison serves the comprehension of the advantages and drawbacks of our proposition. Moreover, it gives us the opportunity to compare a centralized approach with a localized one. In this section we assume that the capacity we computed for RTXP in the previous section is never reached. We first compute relevant parameters of PEDAMACS and then we compare the two protocols on two topologies of 6 nodes.

5.1 Delay, energy and capacity of PEDAMACS

We compute three main parameters, the bound on delay because the solution is real-time, the energy because it is a key issues in WSNs, and the capacity which represents the amount of real-time data the protocol can handle. The notations we use in this section are detailed in table 2.

symbol	signification
k	Network degree
V	Set of vertices
E	Set of edges
$WCTT_{PEDAMACS}$	Theoretical bound on the end-to-end delay
K	Maximum difference of hop-count between two interfering nodes
T_{slot}	Duration of a data slot
$NB_{hops}(p)$	Hop-count of node p
$E_{1hop_PEDAMACS}$	Energy consumed by PEDAMACS for doing one hop
$E_{learning}$	Energy consumed during the learning phase of PEDAMACS
$E_{collection}(p)$	Energy consumed by a node p during during the collection phase of PEDAMACS
$E_{collection}$	Energy consumed by the network during collection phase of PEDAMACS
$D_{max_star_PEDAMACS}$	Maximal end-to-end delay in the case of star network for PEDAMACS
$D_{max_star_RTXP}$	Maximal end-to-end delay in the case of star network for RTXP
$D_{max_line_PEDAMACS}$	Maximal end-to-end delay in the case of linear network for PEDAMACS
$D_{max_line_RTXP}$	Maximal end-to-end delay in the case of linear network for RTXP

Table 2: Notations used in the comparison

For the maximum delay, [10] states that it is ensured that all the packets reach the sink during the scheduling phase (i.e. during the scheduling frame). The maximum length of the scheduling frame depends on the topology, some possible cases are given in [10]. We will consider the case of a general tree graph $G = (V, E)$ with interferences graph being such that the maximum difference of hop-count between two interfering nodes is noted K . Such a graph is retrieved by the sink during the initialization phases as described in section 2.3. In this case the maximum frame length is:

$$WCTT_{PEDAMACS} = (K + 2)(|V| - 1) \times T_{slot}$$

Thus in the case of PEDAMACS the bound on end-to-end delay does not depend on the number of hops.

We now focus on another important parameter in WSNs, the energy consumption. We evaluate the energy-consumption induced by a packet to do one hop. In the case of PEDAMACS it is only the energy used by one node to send the packet and by another to receive it:

$$E_{1hop_PEDAMACS} = E_{snd_packet} + E_{rcv_packet}$$

Based on [10] we can evaluate the energy consumed during the initialization of the network. There is the energy used during the topology learning phase

$$E_{learning} = (V - 1) \times (E_{snd_hello_packet} + k \times E_{rcv_hello_packet})$$

k being the network degree. We also have to take into account the topology collection. We evaluate the energy needed for a node to send its topology packet to the sink:

$$E_{collection}(p) = (E_{snd_hello_packet} + E_{rcv_hello_packet}) \times NB_{hops}(p)$$

We sum this expression for all the nodes in the network.

$$E_{collection} = \sum_{p=1}^{V-1} E_{collection}(p)$$

$E_{collection}$ thus depends on the topology of the network.

The capacity of PEDAMACS is given in [10] for the whole network: each node can send a packet to the sink during a scheduling frame.

5.2 Comparison and discussions

In this section we compare RTXP with PEDAMACS. In order to underline the differences of behavior of the protocols we compare them on two topologies of 6 nodes, a star network and a linear network. In both cases each node has one packet to send.

5.2.1 Star topology

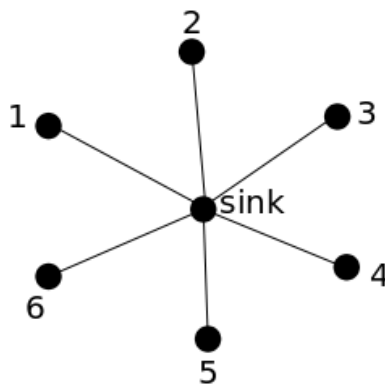


Figure 4: Star network

In the case of the star topology (Figure 4) all the nodes interfere with one another. In the case of PEDAMACS the scheduling is trivial, one slot is reserved for each node to send its packet to the sink. Thus the maximum end-to-end delay for this network is

$$D_{max_star_PEDAMACS} = 6 \times T_{slot}$$

In the case of RTXP, under the capacity limit we are sure that a packet does at least one hop during a duty-cycle, in our case it takes six activity periods thus the maximum delay is

$$D_{max_star_RTXP} = 6 \times D_{activity_period}$$

An activity period is longer than a time slot thus the maximum delay of PEDAMACS is shorter. Nevertheless, the bound on the maximum delay for PEDAMACS is:

$$WCTT_{PEDAMACS} = (K + 2)(|V| - 1) \times T_{slot} = 24 \times T_{slot}$$

It is not tight. The bound of RTXP depends on the duty-cycle ratio

$$WCTT_{RTXP} = NB_{hop_max} \times (D_{activity_period} + D_{sleep})$$

with $NB_{hop_max} = 1$ in this case. In the examples we have the term NB_{hop_max} instead of $(NB_{hop_max} + 1)$ because we are sure all the packets are in the nodes queues at the beginning of the first duty-cycle.

Regarding the energy consumption, we compare the initialization of the two protocols. PEDAMACS initialization is done in two phases, the topology learning and topology collection phases. For RTXP there is just the initialization of the coordinate. From the expressions of energy consumption of the previous sections we deduce that the collection phase of PEDAMACS consumes the same amount of energy as the initialization of the coordinates of RTXP. Thus RTXP consumes

$$E_{collection} = 6 \times (E_{snd_hello_packet} + E_{rcv_hello_packet})$$

less than PEDAMACS for the initialization of the star network. We also deduce that one hop costs more with RTXP than PEDAMACS (with $E_{backoff} = E_{backoff_forward}$):

$$E_{1hop_RTXP} - E_{1hop_PEDAMACS} = 2 \times E_{backoff} + 2 \times E_{snd_jamming}$$

In the star network there are 6 transmissions thus RTXP is better than PEDAMACS in terms of energy consumption if

$$E_{collection} > 6 \times (E_{1hop_RTXP} - E_{1hop_PEDAMACS})$$

5.2.2 Linear topology



Figure 5: Linear network

In the case of linear topology (Figure 5) the nodes two hops away interfere with each others. In this case we use the algorithm given in [10] in order to deduce the schedule. It appears that it is the worst case ($WCTT_{PEDAMACS} = D_{max_line_PEDAMACS}$) thus the bound is tight. In the case of RTXP the bound is also relatively tight:

$$D_{max_line_RTXP} = 5 \times (D_{activity_period} + D_{sleep}) + D_{activity_period}$$

$$\text{and } WCTT_{RTXP} = 6 \times (D_{activity_period} + D_{sleep})$$

By the same reasoning as in the star network case we deduce that RTXP is better than PEDAMACS in terms of energy consumption if:

$$E_{collection} > 21 \times (E_{1hop_RTXP} - E_{1hop_PEDAMACS})$$

because 21 transmissions are needed for all the packets to reach the sink.

5.2.3 Discussion

These two cases show that, even with a fixed topology, it is still not trivial to compare the two protocols. In fact, this comparison highlights the differences between a near-optimal centralized solution and the one we propose, a localized and reliable solution.

In the general case we cannot conclude which protocol has the best end-to-end maximum delay because it depends on the topology. Nevertheless The theoretical bound of RTXP increases with the maximum number of hops to reach the sink, whereas the theoretical bound of PEDAMACS increases with the number of nodes in the network. PEDAMACS is thus more adapted to topologies with few nodes. RTXP is more adapted to networks with few hops to reach the sink but possibly lots of nodes. We can notice that the duty-cycle mechanism is an important parameter of the delay for RTXP. It could be adjusted in order to meet application requirements.

In the case of PEDAMACS the energy consumption for one hop is always less than our solution no matter the time the backoff takes. Nevertheless, PEDAMACS consumes more energy during the initialization (the amount depends on the topology). Moreover the PEDAMACS solution is centralized whereas ours is distributed thus more robust: in PEDAMACS if the sink fails in producing the schedule all the data is lost, in the case of RTXP the failure of a single node does not affect the whole network. Here we highlight a trade-off between energy consumption and robustness.

Another interesting parameter is the packet delivery ratio of these solutions. In both cases, if we only consider packet loss due to collisions the delivery ratio is 100%. We argue that our solution is more reliable in the case of non perfect channel or in the case of node deaths because of the opportunistic nature of the routing part of RTXP. In PEDAMACS, the route is static and there is only one node receiving a given packet at a given time so if it is lost there are no backup solutions. In our solution the packet is received by multiple nodes thus it has more chances to be correctly received by at least one node. This mechanism induces more energy consumption, so there is a trade-off between energy consumption and reliability which has to be done in function of the application.

In RTXP we choose to enhance reliability and robustness at the cost of energy consumption. RTXP is thus more suited to highly critical applications.

PEDAMACS assumes that the sink node can communicate with all the nodes of the network thus the sink node is a special node. In RTXP no assumption is made on the sink node, it is just the node that starts the initialization of the coordinates. Thus in our solution the sink node could be changed during the lifetime of the network in order to tackle the problem of the high energy consumption of the nodes near the sink [24].

We conclude that due to the native differences between PEDAMACS and RTXP it is hard to realize a fair comparison. Nevertheless, RTXP has, in some conditions, worst performances than a centralized one. This shows that a trade-off has to be done when the protocol is chosen between the fact that it is more reliability, robustness and energy consumption or end-to-end delay. As we said RTXP is, to the best of our knowledge, the first hard real-time localized X-layer protocol for WSNs. Indeed, the mechanisms provide hard real-time under the assumption of perfect radio links. We cannot talk about hard real-time with unreliable links because of their probabilistic nature. In the next section we simulate our proposition with both reliable and unreliable links in order to monitor its behavior under such conditions.

6 RTXP performance evaluation

6.1 Simulations parameters

We evaluate the performances of the RTXP by simulation. We use the WSNNet [2] simulator to perform the simulation campaign. In this section we present the simulation parameters and results. We verify the theoretical predictions and test our solution with realistic propagation model. The number of nodes varies from 100 to 800. In each simulation 1000 packets are sent, for each packet the sender is picked randomly in the whole network. The maximal hop-count from the sink in the network depends on the topology, the number of nodes and the propagation model we use. The parameters of the simulation model are detailed in table 3.

parameter	value
number of nodes	100 to 800
bitrate	500kbps
radio range	10 units
area	50×50 units
packet size	100 bytes
jamming code length	200 μ s
propagation model	free space or log-normal shadowing

Table 3: Simulation parameters

6.2 Perfect radio links

First we simulate with the free space model which corresponds only to the path loss without fading or shadowing. This allows to test our algorithm with a perfect channel so we can check theoretical predictions. For all the simulations each point of the curves corresponds to the mean (or maximum value) of 20 simulations with 1000 packets sent in each of them. Figure 6 shows the average delivery ratio in function of the interval of time between packets (i.e. between events detected in the network), so the highest bitrates are at the beginning of the abscissa axis on Figure 6, the higher the interval between packet (inter arrival), the lower the bitrate. The results are showed for networks of 300 nodes. Figure 6 shows that for an inter arrival of more than 0.2 seconds the delivery

ratio is 100%. It means that the capacity C_{RTXP} that we calculated in section 4.1 is never reached. Nevertheless, in some special topology cases (lots of nodes in the same area) it could be the case. For the rest of the simulations we choose an inter arrival of packets of 1 second so we are sure to be under the capacity limit.

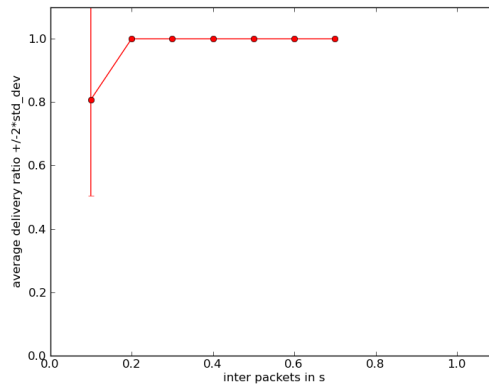


Figure 6: Average delivery ratio in function of inter packets for a network of 300 nodes

In the rest of the simulations with perfect links we verify that the delivery ratio is always 100%. Figure 7 shows that the theoretical bound on packet end-to-end delay has never been reached during any simulation (the theoretical bound is calculated with the formula of section 4.2 for the maximum number of hops of the simulations). We also notice that the average end-to-end delay is about one fourth of the theoretical bound. In the case of nodes deaths (because they empty their batteries), there are two cases. Either the coordinate is still coherent, which means that the nodes are still connected with their upper and lower ring, or it is not the case. In the first case the protocol behaves the same as in the cases where the network density is lower. In the second case the coordinate has to be updated.

6.3 Realistic radio links

Figure 8 represents the same curves as Figure 7 with a log-normal shadowing model. The log-normal shadowing model has been proven [31] to be very suited to model real wireless links in the case of WSN. The path loss plus a random variable are taken into account. The maximum delay, in this case, never reaches the theoretical bound. And the ratio between the average and the maximum values is in the same order of magnitude as in the simulations with free-space propagation model.

Nevertheless in this case the delivery ratio is not always 100%. This is due to the fact that some packets and jamming codes are lost during the process. If a node which has sent a packet does not receive a jamming code from any forwarder it notifies it in the L slot and contends in the next activity period. Here we choose to drop the packets which are late. We can notice that it may

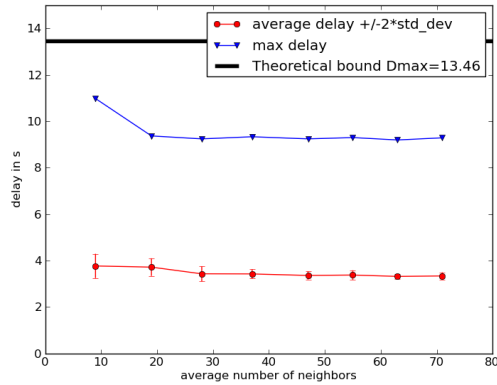


Figure 7: Average and maximum delay in function of network density compared to theoretical bound for a 5 hops network

be an issue with applications that need a very high reliability. Figure 9 shows that for low density networks the delivery ratio more than 80% and it grows slightly with the network density. The growth can be explained because the more potential forwarder there is the more probable the success of the election of the forwarder is.

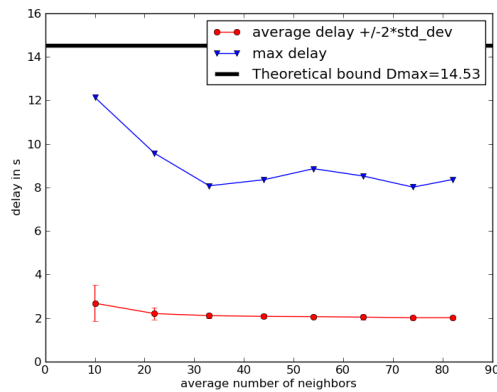


Figure 8: Average and maximum delay in function of network density compared to theoretical bound for a 3 hops network with log-normal shadowing

In this section we verify the theoretical predictions done in the previous sections and we show that our protocol can be used on an unreliable link.

7 Conclusion and future work

In this document we present a solution to realize real-time communications in a WSN under a set of hypothesis. The solution is based on a VCS which properties

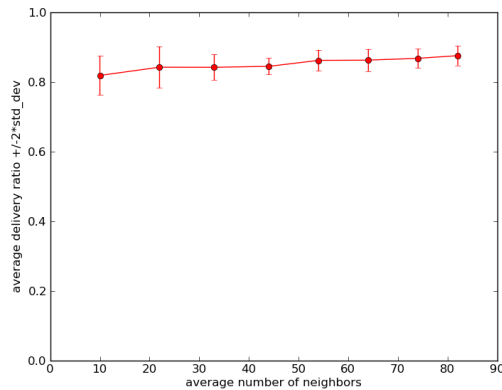


Figure 9: Average delivery ratio in function of network density with log-normal shadowing

are to classify the nodes in a hop-count ring and give them an unique identifier in a 2 hops neighborhoods. We propose RTXP a real-time X-layer protocol that guarantees a bound on the end-to-end delay of packets in the network. We present the parameters of our solution by comparing it to a centralized solution as our protocol is, up to our knowledge, the first localized hard real-time protocol for WSNs. We discuss the advantages and drawbacks of both. We conclude that there is a trade-off between reliability and energy consumption and maximum delay. We then evaluate the performances of our solution by simulation. The results show that the deadline is always met during the simulations. We also conclude that RTXP is reliable (about 80% delivery ratio) with a realistic radio link under harsh conditions, but it does not guarantee the delivery of all the packets and thus the hard real-time property is lost. This can be an issue in highly critical applications. Two solutions can be explored to address this issue, first work can be done at the physical layer in order to make the radio links more reliable by optimizing decoding thresholds for example (selection of the best links while keeping connectivity). Another solution could be an algorithm which reserves a good path in terms of links quality from the source to the sink, but this would imply more signalization thus more energy consumption.

The protocol has to be formally validated. Finally experimentation on real sensors has to be performed in order to verify the performances of our solution.

References

- [1] Cc2500, low-cost low-power 2.4 ghz rf transceiver.
- [2] <http://wsnet.gforge.inria.fr/>.
- [3] R. Akl and Y. Saravanos. Hybrid energy-aware synchronization algorithm in wireless sensor networks. PIMRC '07, pages 1–5, Athens, Greece, 2007.

-
- [4] A. Bachir, D. Barthel, M. Heusse, and A. Duda. Micro-frame preamble mac for multihop wireless sensor networks. ICC '06, pages 3365–3370, Istanbul, Turkey, 2006.
 - [5] Michael Buettner, Gary V. Yee, Eric Anderson, and Richard Han. X-mac: A short preamble mac protocol for duty-cycled wireless sensor networks. SenSys '06, pages 307–320, Boulder, USA, 2006.
 - [6] Marco Caccamo, Lynn Y Zhang, Lui Sha, and Giorgio Buttazzo. An implicit prioritized access protocol for wireless sensor networks. RTSS '02, pages 39–48, Austin, USA, 2002.
 - [7] Qing Cao and T. Abdelzaher. A scalable logical coordinates framework for routing in wireless sensor networks. RTSS '04, pages 349–358, Lisbon, Portugal, 2004.
 - [8] Jianrong Chen, Peidong Zhu, and Zhichang Qi. Pr-mac: Path-oriented real-time mac protocol for wireless sensor network. ICESS '07, pages 530–539, Daegu, Korea, 2007.
 - [9] Jiming Chen, Ruizhong Lin, Yanjun Li, and Youxian Sun. Lqer: A link quality estimation based routing for wireless sensor networks. In *Sensors*, volume 8, pages 1025–1038, 2008.
 - [10] S. Coleri. Pedamacs: Power efficient and delay aware medium access protocol for sensor networks, ms thesis, electrical engineering and computer science, university of california, berkeley, December 2002.
 - [11] Lance Doherty and Kristofer S. J. Pister. Tsmc: Time synchronized mesh protocol. DSN '08, pages 391–398, Orlando, USA, 2008.
 - [12] M. Dohler, D. Barthel, F. Maraninchi, L. Mounier, S. Aubert, C. Dugas, A. Buhrig, F. Paugnat, M. Renaudin, A. Duda, M. Heusse, and F. Valois. The aresa project: Facilitating research, development and commercialization of wsns. SECON'07, pages 590–599, 2007.
 - [13] S.C. Ergen and P. Varaiya. Pedamacs: power efficient and delay aware medium access protocol for sensor networks. In *IEEE Transactions on Mobile Computing*, volume 5, pages 920–930, 2006.
 - [14] Emad Felemban, C.G. Lee, and E. Ekici. Mmspeed: Multipath multi-speed protocol for qos guarantee of reliability and timeliness in wireless sensor networks. In *IEEE Transactions on Mobile Computing*, volume 5, pages 738–754, 2006.
 - [15] Pei Huang, Hongyang Chen, Guoliang Xing, and Yongdong Tan. Sgf: A state-free gradient-based forwarding protocol for wireless sensor networks. In *ACM Trans. Sen. Netw.*, volume 5, pages 14:1–14:25, 2009.
 - [16] Brad Karp and H. T. Kung. Gpsr: greedy perimeter stateless routing for wireless networks. MobiCom '00, pages 243–254, Boston, USA, 2000.
 - [17] Kavi Kumar Khedo, Rajiv Perseedoss, and Avinash Mungur. A wireless sensor network air pollution monitoring system. In *International Journal of Wireless and Mobile Networks (IJWMN)*, Vol.2, No.2, pages 31–45, 2010.

-
- [18] G. Lu, B. Krishnamachari, and C.S. Raghavendra. An adaptive energy-efficient and low-latency mac for data gathering in wireless sensor networks. IPDPS '04, page 224, Santa Fe, USA, 2004.
- [19] Alexandre Mouradian and Isabelle Augé-Blum. 1-D Coordinate Based on Local Information for MAC and Routing Issues in WSNs. Research report RR-7819, INRIA, November 2011.
- [20] Joseph Polastre, Jason Hill, and David Culler. Versatile low power media access for wireless sensor networks. SenSys '04, pages 95–107, Baltimore, USA, 2004.
- [21] Maneesha V. Ramesh. Real-time wireless sensor network for landslide detection. In *SENSORCOMM '09*, pages 405–409, Athens, Greece, 2009.
- [22] Ananth Rao, Sylvia Ratnasamy, Christos Papadimitriou, Scott Shenker, and Ion Stoica. Geographic routing without location information. MobiCom '03, pages 96–108, San Diego, USA, 2003.
- [23] Utz Roedig, Andre Barroso, and Cormac J. Sreenan. f-mac: A deterministic media access control protocol without time synchronization. EWSN '06, pages 276–291, Zurich, Switzerland, 2006.
- [24] Leila Ben Saad and Bernard Tourancheau. Towards an efficient positioning of mobile sinks in wireless sensor networks inside buildings. NTMS '09, pages 1–5, Cairo, Egypt, 2009.
- [25] Sergio D Servetto and Cornell Univerisity. Constrained random walks on random graphs : Routing algorithms for large scale wireless sensor networks. WSN '02, pages 12–21, Atlanta, USA, 2002.
- [26] J.a. Stankovic and T. Abdelzaher. Speed: a stateless protocol for real-time communication in sensor networks. ICDCS '03, pages 46–55, Providence, USA, 2003.
- [27] Rui Tan, Guoliang Xing, Jinzhu Chen, Wen-Zhan Song, and Renjie Huang. Quality-driven volcanic earthquake detection using wireless sensor networks. RTSS '10, pages 271–280, San Diego, CA, USA, 2010.
- [28] T. Watteyne, I. Augé-Blum, M. Dohler, and D. Barthel. Geographic forwarding in wireless sensor networks with loose position-awareness. PIMRC '07, pages 1–5, Athens, Greece, 2007.
- [29] Thomas Watteyne, Isabelle Augé-Blum, and Stéphane Ubéda. Dual-mode real-time mac protocol for wireless sensor networks: a validation/simulation approach. InterSense '06, page 2, Nice, France, 2006.
- [30] Wei Ye, John Heidemann, and Deborah Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. In *IEEE/ACM Trans. Netw.*, volume 12, pages 493–506, 2004.
- [31] Marco Zuniga and Bhaskar Krishnamachari. Analyzing the transitional region in low power wireless links. SECON '04, pages 517–526, Santa Clara, USA, 2004.

Contents

1	Introduction	3
2	Related work	4
2.1	Medium Access Control	4
2.2	Routing	6
2.3	Cross-layer	7
3	Hypothesis and problem statement	8
3.1	Hypothesis	8
3.2	Problem statement	9
4	Proposition: a novel Real-Time X-layer Protocol, RTXP	9
4.1	General idea	9
4.2	Virtual coordinate system	10
4.3	In-depth detail of RTXP	10
4.4	Capacity, delay and energy	13
4.5	Trade-offs	15
5	Theoretical study: comparison between RTXP and PEDAMACS mechanisms	15
5.1	Delay, energy and capacity of PEDAMACS	15
5.2	Comparison and discussions	17
5.2.1	Star topology	17
5.2.2	Linear topology	18
5.2.3	Discussion	19
6	RTXP performance evaluation	20
6.1	Simulations parameters	20
6.2	Perfect radio links	20
6.3	Realistic radio links	21
7	Conclusion and future work	22



**RESEARCH CENTRE
GRENOBLE – RHÔNE-ALPES**

Inovallée
655 avenue de l'Europe Montbonnot
38334 Saint Ismier Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399