AreaCast: a Cross-Layer Approach for a Communication by Area in Wireless Sensor Networks

Karel Heurtefeux
CNRS, Verimag
UMR 5104, Grenoble
F-38041, France
Email: karel.heurtefeux@imag.fr

Florence Maraninchi
Grenoble INP, Verimag
UMR 5104, Grenoble
F-38041, France
Email: florence.maraninchi@imag.fr

Fabrice Valois
Université de Lyon, INRIA
INSA-Lyon, CITI
F-69621, France
Email: fabrice.valois@insa-lyon.fr

***Abstract*—To provide for reliability in Wireless Sensor Networks (WSNs), Medium Access Control (MAC) protocols must be adapted by mechanisms taking cross-layer approaches into account. We describe AreaCast which is designed for enhancing reliability in WSNs. AreaCast is a MAC layer mechanism independent of the routing layer, but uses only local topological and routing information to provide a communication *by area* instead of a traditional, *node-to-node* communication (i.e. unicast). In AreaCast, a source node addresses a set of nodes: an *explicit relay* node chosen as the next hop by a given routing protocol, and k other *implicit relay* nodes. The neighboring nodes select themselves as implicit relays according to their location from the explicit relay node. This mechanism uses overhearing to take advantage of the inherent *broadcast* nature of wireless communications. Without changing the routing protocol, AreaCast is able to dynamically avoid a byzantine node or an unstable link, allowing to benefit from the inherent topological redundancy of densely deployed sensor networks. Simulation results show that AreaCast significantly improves the packet delivery rate while having a good reliability-energy consumption trade-off.**

I. Introduction

Wireless Sensor Networks (WSNs) are formed by hundreds or thousands of low-cost and low-energy sensor devices. Sensors are densely deployed over a geographic area to collect data. Such networks have a wide range of potential applications: ambient home, smart building, environmental monitoring, body space... Despite intensive research efforts, numerous challenges remain for large-scale deployment in real environments. WSNs are more vulnerable than conventional wireless and wired networks. Sharing the wireless medium in an energy-efficient way is a key point in WSNs. As a result, MAC layer design is crucial in such a context.

Contributions. In this paper, we propose a new MAC layer mechanism, AreaCast. It uses routing-layer information (distance, neighborhood and route information), to enhance robustness, but it does not change the routing protocol. AreaCast uses the selected node given by a routing protocol (called *explicit relay*) but also k *implicit relay* nodes within an area close to it (Fig. 1). If the explicit relay is unable to fulfill its role, implicit relays take its place. In this way, the network redundancy is better exploited to provide robustness. Note that AreaCast is independent of the routing process and does not influence the route construction. AreaCast increases the delivery rate in an energy-efficient manner compared to Sensor-MAC [1]. Simulations are performed for faulty nodes, volatile links and a realistic propagation model.

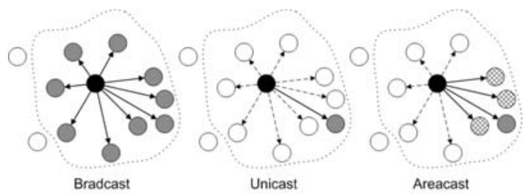


Fig. 1: Broadcast, Unicast and AreaCast

Structure of the paper. The rest of the paper is organized as follows. Section II provides an overview of MAC protocols designed for WSNs. In Section III, we discuss weaknesses of unicast in MAC protocols and we describe AreaCast. In Section IV, the AreaCast performances are presented in terms of energy consumption and delivery rate. Finally, Section V contains concluding remarks and some future work directions.

II. Problem Statement

The MAC layer coordinates access to a medium, common to several processes. It has a central role in any communication system and its behavior has an important impact on the WSNs performances. As a result, the concurrent access to the communication channel in wireless networks has been extensively studied for both ad-hoc and sensor networks. The radio is the main cause of power consumption. The design of MAC protocols is crucial to enhance WSN lifetime. Because of open medium, dynamic topology, absence of central control, and constrained capabilities, sensor networks are vulnerable and prone to interferences and hardware failures.

How to share the wireless medium to provide robustness (i.e. deal with failures or collisions) in an energy-efficient way? That is the problem we address in this paper.

A. State of the art

The main MAC layer protocols designed for WSNs proposed in literature address the energy issue. But some studies

978-1-4577-1826-7/11/$26.00 ©2011 IEEE

introduce cross-layer (MAC and routing) or just routing protocols using the network redundancy to enhance the robustness.

The authors in [2] present a cross-layer protocol called SPEED. SPEED uses geographic location to provide a soft real-time communication service and to handle congestion. Thus, the main concern of SPEED is the end-to-end delay. The Stateless Non-deterministic Geographic Forwarding algorithm (SNGF) defined in this paper, uses, like AreaCast, the notion of area in the forwarding process. A source node determines a forwarding candidate set among its neighbors according their geographic position. But it sends a packet to a single node belonging to this set. This forwarding node is chosen according a metric based on optimal path length and load balancing. Unlike AreaCast, broadcast nature of wireless communications is not exploited and if the forwarding node chosen fails, a retransmission is needed.

The authors in [3] present a cross-layer protocol called ExOR. The goal of this protocol is to improve the throughput by using long radio link, generally avoided by traditional routing. ExOR chooses a forwarding node in two steps. It first broadcasts a packet and then chooses a receiver to forward after learning the set of nodes which actually received the packet. The source node includes in each packet a list of candidate forwarders prioritized by closeness to the destination. This solution is clearly dependent to the routing protocol and increases size of packets exchanged.

The authors in [4] propose a resilient packet-forwarding scheme using overhearing of the neighbors. Our aim is similar, however, they only consider the routing layer and their solution duplicates a packet to create multi-path data forwarding when they detect relaying nodes' misbehavior. Traffic redundancy leads to an important waste of energy. Our solution considers the MAC layer and is independent of routing protocols.

The Geocast protocol [5] proposes a routing and addressing method to integrate geographic coordinates into Internet Protocol. Geocast enables the creation of location dependent service. Based on this method, many new protocols or improvements have been developed [6], [7]. Geocast and Geocast-based protocols need GPS coordinates. However, dedicated hardware like GPS is not always suitable in embedded systems like sensor. Moreover, the proposed solutions do not tackle MAC layer or robustness against faulty links or nodes. Finally, Geocast is a form of specific multicast addressing.

B. Motivations

Our solution considers the MAC layer level and is independent of a given routing protocol. However, AreaCast is based on a local knowledge of the topology and route information (next hop). It is able to make a decision without GPS information. AreaCast can be considered as a cross-layer approach because of the need of this network-layer information. By exploiting redundancy, it improves the end-to-end reliability. Finally, AreaCast is original by tackling two crucial issues: robustness and energy saving.

III. The AreaCast Protocol

Unicast in MAC protocols and its weaknesses. From our point of view, unicast addressing is particularly not suitable for dense and fragile sensor networks. When a node or a link disappears, MAC protocols unsuccessfully try to retransmit packets, instead of exploiting the natural topological redundancy. Even if a node is close to the relay node, a traditional MAC protocol considers only the latter. Before selecting another relay node, a source node tries to reach the same node until the retry limit. These retransmissions result in an important waste of energy, a source of packet loss and an increase of end-to-end delays.

Protocol Overview. To avoid useless retransmissions, AreaCast proposes a new communication pattern. In WSNs, the identity of relay nodes is useless in the multi-hop routing process. But this identity is still used to address a particular neighboring node chosen by a given routing protocol as the next hop. In AreaCast, a source node addresses an *area* instead of addressing only one node in an unicast manner. The *area* is composed by the *explicit relay* node (the next hop) and k self-elected *implicit relay* nodes. If the *explicit relay* node does not respond, an *implicit relay* node takes its place in the multi-hop routing in a transparent way. The Figure 2 shows how *explicit relay* nodes can be used to bypass the faulty nodes.

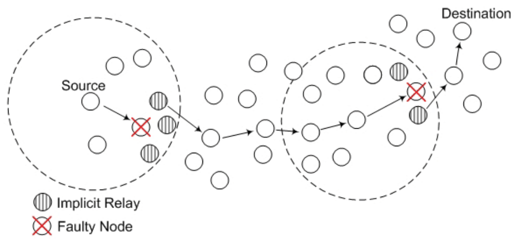


Fig. 2: An example of AreaCast process. Grey nodes represent implicit relays using to bypasses faulty nodes.

To address an area instead of one node, several difficulties have to be considered: the election of the implicit relay nodes (the criteria of selection and the number of implicit relay nodes) and the backoff duration among the implicit and explicit relay nodes to avoid collisions.

The election of implicit relay nodes is crucial for both energy consumption and relay efficiency. If AreaCast selects numerous implicit relay nodes, the probability of reception is improved, however, the overhearing and the energy consumption are also increased. Moreover, the backoff algorithm of each implicit relay needs a special attention to avoid collisions.

Firstly, we describe the election method of implicit relay nodes, and secondly, we introduce the AreaCast protocol applied to a sleep/listen approach. Finally, we discuss the AreaCast application in the context of preamble frame protocols.

Criteria for implicit relay nodes election. Limiting the number of *implicit relay* nodes is necessary for several reasons: the energy consumption, the end-to-end delay and the relay efficiency.

Here, the maximal number of *implicit relay* nodes is limited to three to provide enough redundancy without impacting too much the delay and the energy consumption. The optimal value of k will be addressed in a future paper.

Each node selects itself, or not, as a implicit relay according several information: local topology (1 and 2-hop neighborhood and *distance* between its neighbors) and route next-hop. This information are obtained by exchanging periodic *hello* packets and by the reception or RTS/CTS packets (for the route next-hop). In this way, each neighbor of the explicit relay node knows *distance* between the explicit and potential implicit relays. Therefore, the neighboring nodes have the same uniform view and they are able to establish the same ranking. They elect themselves as implicit relays without exchanging extra packets.

The election algorithm of a node as implicit relay is described as follow:

- A node X is the explicit relay node if it is addressed as the next hop by a source node
- A node X elects itself as an implicit relay if and only if:
  - X is not the *explicit relay* node
  - X is a neighbor of the source node
  - X is a neighbor of the *explicit relay* node
  - X is one of the first k nodes in the ranking established with the following rules:
    - At the first ranks, the neighbors of the second next hop are placed according to their distance from the explicit relay.
    - At the following ranks, the nodes which are not neighbors of the second next hop are placed according to their distance from the explicit relay.

In other words, the priority is given to the nodes which are neighbors, at the same time, of the source, the explicit relay and the second next hop.

Distance criterion. The main criterion used in implicit relay election is the distance from the explicit relay node. This metric can be determined in various manners. A quantitative way is available, if each node has GPS information [8] or using RSSI (even if the ineffectiveness of RSSI is mentioned in the literature [9]). Or in a qualitative way, by using alternative protocols such as QLoP [10]. The quantitative distance is computed based on physical measures and is meant to be close to the real geographical distance. The quantitative distance protocols generally do not take into account the energy consumption and assume that each node is able to compute easily the time or the angle of arrival. The quantitative distance defined in QLoP is not directly connected to the real distance but computes a proximity indicator between nodes. The quantitative distance protocols use only topological information. Moreover, because QLoP provides a ranking between neighboring nodes according to their proximity, it is particularly suitable for AreaCast.

AreaCast Applications. In this section, we describe the application of AreaCast with $k = 3$ in a sleep/listen duty cycle protocol. A node wishing to send data initiates the process by sending a RTS frame. This frame is broadcasted to all neighboring nodes. The destination identity and transmission time are included in the frame. This indicates to other nodes that they should refrain from sending data at the same time. When neighboring nodes receive a RTS packet, they self-elect or not as implicit relay node according to the criteria given above. The implicit relay nodes stay awake while the other neighboring nodes go to sleep during the communication time (Algorithm 1).

If the explicit relay node does not send a CTS frame during the given time t_0, the first implicit relay node sends a CTS frame to the source node. If the first implicit relay node fails, the second implicit relay sends CTS at t_1, etc. If none of the relay nodes succeeds in the CTS sending, the source node retransmits the RTS packet. If one relay succeeds in the CTS sending, the following relay nodes cancel their backoff. Note that each relay, implicit and explicit, has its proper backoff timer to transmit: t_0 is reserved to the explicit relay node response, and times t_1,t_2,t_3 to the response of first, second and third implicit relay nodes respectively ($t_0 < t_1 < t_2 < t_3$ are fixed). Moreover, because relay nodes are neighbors of the source, they are able to cancel their backoff timer if the communication is not disturbed.

When a source node receives a CTS packet, from the explicit or an implicit relay node, it sends the data packet to the explicit relay node (Algorithm 2). When a node receives a data packet, the behavior is similar to the reception of a RTS: implicit relay nodes listen to the channel to know if the explicit relay node responds an ACK frame. If not, they dynamically replace it (Algorithms 3 and 4). AreaCast allows avoiding the useless retransmissions and saves energy and time.

Algorithm 1 Node X Receives RTS

if X is explicit relay **then**
    Send CTS
else
    if $X \in$ the 3 closest neighbors of the explicit relay node **then**
        Elect itself as implicit relay and chooses its backoff timer
    else
        sleep
    end if
end if
if X is implicit relay $\wedge$ backoff timer expired **then**
    send CTS to Source
end if

Algorithm 2 Node X Receives CTS from node Y

if X is implicit relay **then**
    cancel backoff timer
else
    if X is Source **then**
        Send DATA to Y
    end if
end if

Note that AreaCast does not involve redundant messages. When a relay node (explicit or implicit) forwards a message, thanks to overhearing, the other relay nodes are aware and do not forward the same message.

The application of AreaCast to preamble-based protocols is quite simple. Implicit relay nodes self-elect when they receive

Algorithm 3 Node X Receives DATA from source node

```
if X is explicit relay then
    send ACK to source node
else
    if X is implicit relay then
        chooses its backoff timer
    end if
end if
if X is implicit relay ∧ backoff timer expired then
    send ACK to Source
end if
```

Algorithm 4 Node X Receives ACK from node Y

```
if X is implicit relay then
    cancel backoff timer
end if
```

the preamble frame (identity of the next hop is included). They stay awake during the data packet transmission and send an ACK packet according to the behavior of the explicit relay node.

IV. Evaluation

All the results provided in this section were obtained using WSNet, an event-driven simulator for wireless networks.

Simulation parameters. Nodes are randomly deployed on a plane square and are motionless. Each node periodically sends a `hello` packet to discover neighborhood, builds and maintains a logical structure through the network. In our simulation, a unique sink is assumed at the center of the field. A central shortest path routing protocol is considered. To illustrate the dynamicity and weakness of a WSN, we consider p faulty nodes randomly and uniformly distributed on the network (p varies between 0% and 50% of the node population). We also consider the probability q of faulty links (q varies between 0 and $1/2$). The focus of our simulations is on comparing the AreaCast protocol with the S-MAC protocol. Note that S-MAC uses a classical approach (CSMA-CA with RTS/CTS exchange) to rule the contention access. The results are averaged over 100 simulation runs for each case with a 95% confidence interval. The number k of implicit relay nodes is fixed to 3.

The results are averaged over 100 simulation runs for each case with a 95% confidence interval. Table I sums up the simulation parameters.

Parameter	value	
Number of nodes	100	
Field size	$100 \times 100m$	
Propagation model	ideal with no collision nor interference	realistic with log-normal shadowing propagation model
Transmission range	$20m$	
Standard deviation		$2dB$
Transmission power		$-30dBm$
Pathloss exponent		2
Node failures probability	0% to 50%	
Link failures probability	0% to 50%	
Simulated time	$40s$	
Simulation time	$30 - 40s$	
Number of runs	100	

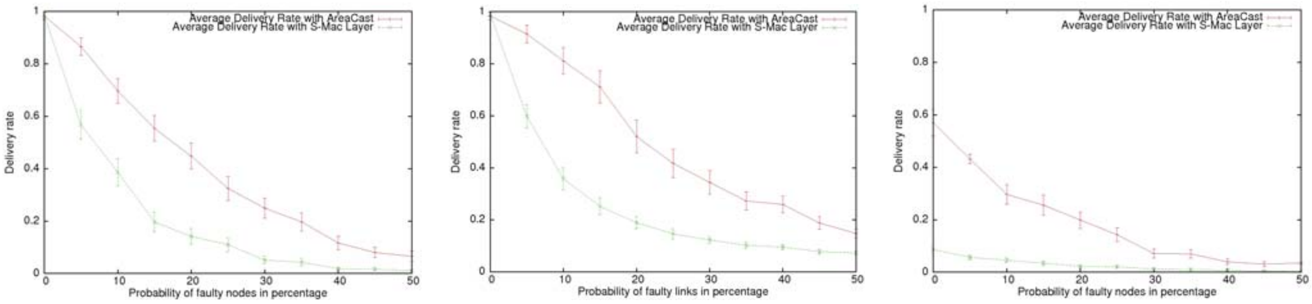
TABLE I: Summary of the simulation parameters.

Propagation model. We consider two propagation models: In the first one, ideal, two processes u and v can communicate if and only if their Euclidean distance is at most rad, where rad is the transmission range. In this ideal propagation model, there are neither interferences nor collisions. In the second model, realistic, the range of a radio system is based upon the definition of a signal-to-noise ratio (SNR) threshold and it models interferences. It should be noted that this assumption leads to a neighborhood instability and links between nodes can be uni-directional.

Energy consumption model. When modeling energy consumption, there are essentially two approaches: (i) indirect modeling, in which global assumptions like "*sending a message to the sink costs k units of energy*" have to be made; in this case, evaluating energy consumption amounts to counting messages; the validity of the assumptions may be hard to assess; (ii) direct modeling of the consuming hardware elements like the radio device (usually in the form of *power-state* models), coupled with the description of the software that drives them. The latter option is the one implemented in *emulators*, where the details of the execution platform are represented. Counting messages is too abstract because the idle listening periods (when a node listens to an idle channel to receive potential traffic) are not taken into account, although they contribute to the overall energy consumption in a significant manner.

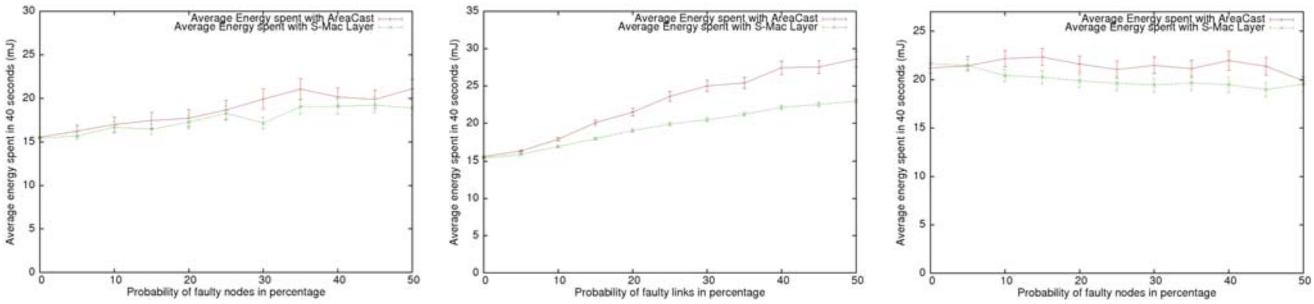
The approach we follow here is based on an explicit modeling of the power states of the radio device. But it is more abstract than emulators, to preserve good simulation times. The model of the radio is a 4-state automaton (Fig. 6). Each state represents a consumption *mode*: sleep, idle, receive, or send. Each state is associated with a value related to the instantaneous energy consumption while the radio is in the corresponding mode. The energy consumption labels are taken from the datasheet of the TI CC1100 [11] radio device. The Sleep mode has the lowest consumption; the radio is not able to transmit nor receive. The idle mode is the default state when the radio is not receiving nor transmitting. The receive mode is when the radio is receiving or listening on the wireless channel. The send mode is when the radio is transmitting. To evaluate the total energy consumption of a node during a given scenario, one has to keep track of the time spent in each of the modes. For this we need to relate the current state of the MAC protocol to the current state (mode) of the radio device.

In this experiment, we consider that the MAC protocol controls the mode changes of the radio entirely. This means that we ignore the situations where the MAC protocol issues a command to the radio to reach a given state (e.g., transmit) but the radio takes some time to get there (e.g., because of a calibration process). Ignoring these intermediate states is allowed if their duration is sufficiently short. When the MAC controls the mode changes entirely, it is sufficient to track the states of the MAC protocol, to be able to track the states of the radio, and hence to compute the total energy consumption. Our simulation approach combines the precision of a direct



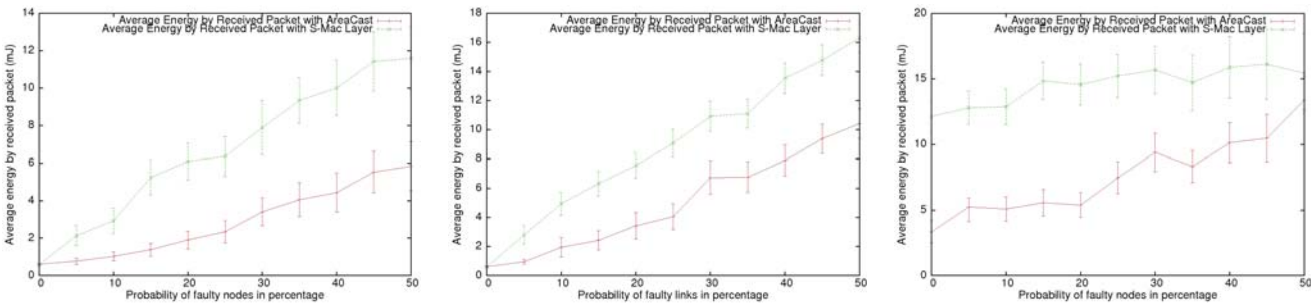
(a) (b) (c)

Fig. 3: Evolution of delivery ratio: (a) against node failures; (b) against link failures; (c) against node failures in case of realistic propagation model



(a) (b) (c)

Fig. 4: Protocol overcost in terms of total energy consumption: (a) against node failures; (b) against link failures; (c) against node failures in case of realistic propagation model



(a) (b) (c)

Fig. 5: Energy consumption per received packet: (a) against node failures; (b) against link failures; (c) against node failures in case of realistic propagation model

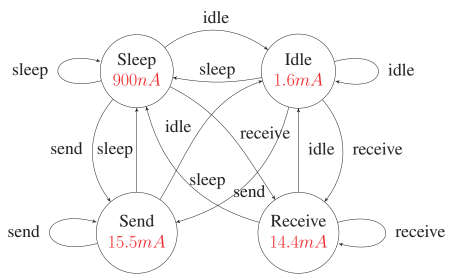


Fig. 6: Radio modeled by a finite-state automaton

modeling of energy consumption, with the performances of abstract simulations. It is integrated in WSNET.

Evaluation metrics. To determine performances of the two compared MAC protocols, we measure the average delivery ratio (ratio between the total number of sent packets and the total number of received packets). This metric allows us to measure the gain in efficiency between the two MAC protocols. Average percentage of packets forwarded by at least one implicit relay node is also measured. In this study, we investigate in particular energy consumption: the total energy consumed by nodes and energy consumption per received packet (expressed in mJ) are evaluated. The energy consumption is computed according to time spent in different states.

Results and analysis. As expected, the average delivery ratio (Fig. 3) decreases when increasing the number of faulty nodes or links. When the AreaCast protocol is considered, the delivery ratio is significantly higher compared to S-MAC. Using implicit relay nodes allows to continue a RTS/CTS/DATA/ACK dialog at any moment. In S-MAC, the probability of a successful communication is the probability to transmit successfully RTS, CTS, DATA and ACK packets. If one of this communication fails, a retransmission is needed. While in AreaCast, at least one implicit node continues the communication in a dynamic manner. With a realistic propagation model, the delivery ratio of the S-MAC protocol is low even in the absence of faulty nodes. The shortest-path routing algorithm favors distant relay nodes and therefore, weak links. The probability to lose packets increases when the number of hops increases. Moreover, the retransmission mechanism used in the S-MAC protocol increases interferences and collisions. AreaCast is able to handle part of the traffic to the sink using implicit relay nodes to bypass faulty nodes and links. Finally, with AreaCast the network continues to operate even in presence of faulty nodes and links.

On the one hand, because of overhearing, AreaCast has an energy consumption overcost. Nevertheless, this overhearing concerns a small part of the network nodes involved in the multi-hop routing. On the other hand, it minimizes the number of retransmissions and therefore the energy consumption. Moreover, with S-MAC, since dead nodes or faulty links lead to losing packets, this saves energy. As a result, the total energy consumption difference between AreaCast and S-MAC increases when the probability of faulty nodes or links increases (Fig. 4). However, when we study average energy spent per received packet, we note a clearly less important energy consumption. The gain is really important when the number of faulty nodes or links is high. This signifies that AreaCast is a good trade-off between energy consumption and network reliability (Fig. 5).

V. Conclusions and further work

In this article, we have proposed a MAC mechanism enhancing the reliability under realistic signal propagation model and in presence of faulty nodes and links. The mechanism uses overhearing and information from routing layer to elect k *implicit relay* nodes within an area close to the *explicit relay* node. AreaCast protocol is especially designed to WSNs, where density is important and nodes are prone to failures. The communication *by area* dynamically avoids faulty nodes and unstable links. Note that the AreaCast protocol is independent of a given routing protocol and do not involve redundant messages. Our simulations show that, despite the increasing number of faulty nodes or links, the network is able to continue to deliver data packets to the sink while keeping a satisfactory energy-reliability trade-off.

Acknowledgment

This work is partially supported by the French Ministry of Research under contract ANR VERSO ARESA2 2009-2012.

References

[1] W. Ye, J. Heidemann, and D. Estrin, “Medium access control with coordinated adaptive sleeping for wireless sensor networks,” *IEEE/ACM Transactions on Networking*, vol. 12, pp. 493–506, 2004.

[2] T. He, J. A. Stankovic, C. Lu, and T. Abdelzaher, “Speed: A stateless protocol for real-time communication in sensor networks,” *Distributed Computing Systems, International Conference on*, vol. 0, p. 46, 2003.

[3] S. Biswas and R. Morris, “Exor: opportunistic multi-hop routing for wireless networks,” in *Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*, ser. SIGCOMM ’05. New York, NY, USA: ACM, 2005, pp. 133–144. [Online]. Available: http://doi.acm.org/10.1145/1080091.1080108

[4] S.-B. Lee and Y.-H. Choi, “A resilient packet-forwarding scheme against maliciously packet-dropping nodes in sensor networks,” in *Proceedings of the fourth ACM workshop on Security of ad hoc and sensor networks*, ser. SASN ’06. Alexandria, USA: ACM, October 2006, pp. 59–70. [Online]. Available: http://doi.acm.org/10.1145/1180345.1180353

[5] T. Imielinski and J. C. Navas, “Gps-based addressing and routing,” RFC2009, Tech. Rep., 1996.

[6] I. Stojmenovic, “Geocasting with guaranteed delivery in sensor networks,” *IEEE Wireless Communications*, vol. 11, pp. 29–37, 2004.

[7] A. B. Bomgni, J. F. Myoupo, and A. O. Cheikhna, “Randomized multi-stage clustering-based geocast algorithms in anonymous wireless sensor networks,” in *Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*, ser. IWCMC ’09. New York, NY, USA: ACM, 2009, pp. 286–291. [Online]. Available: http://doi.acm.org/10.1145/1582379.1582442

[8] B. Hofmann-Wellenhof, H. Lichteneeger, and J. Collins, *Global Positioning System: Theory and Practice*. Springer Wien New York, 2001.

[9] P. Kumar, L. Reddy, and S. Varma, “Distance measurement and error estimation scheme for rssi based localization in wireless sensor networks,” in *Wireless Communication and Sensor Networks (WCSN), 2009 Fifth IEEE Conference on*, december 2009, pp. 1 –4.

[10] K. Heurtefeux and F. Valois, “Distributed qualitative localization for wireless sensor networks,” in *7th International Conference on AD-HOC Networks & Wireless (ADHOC-NOW)*, Sophia Antipolis, France, September 2008.

[11] “Chipcon inc., cc1100 datasheet,” http://www.chipcon.com/, 2008.