
Volume 30, issue 13, 26 September 2007
ISSN 0140-3664
ELSEVIER
computer
communications
Available online at
ScienceDirect
www.sciencedirect.com
Special Issue:
Sensor-Actuated Networks
Guest Editors: Mohamed Eltoweissy
Silvia Giordano
Stephan Olariu
David Simplot-Ryl

Available online at www.sciencedirect.com
ScienceDirect
ELSEVIER
Computer Communications 30 (2007) 2603–2614
computer communications
www.elsevier.com/locate/comcom

$O(1)$-Reception routing for sensor networks

Abdelmalik Bachir [a,b,*], Dominique Barthel [b], Martin Heusse [a], Andrzej Duda [a]

[a] France Telecom R&D, Meylan, France
[b] LSR-IMAG Laboratory, Grenoble, France

Available online 27 June 2007

Abstract

In traditional approaches to energy-efficient routing, a node needs to receive routing messages from all of its neighbors to be able to select the best route. In this work, we propose a technique that enables the best route selection based on exactly one message reception. We call the resulting routing protocol $O(1)$-reception. In $O(1)$-reception, each node delays forwarding of routing messages (RREQs) for an interval inversely proportional to its residual energy. This energy-delay mapping technique makes it possible to enhance an existing min-delay routing protocol into an energy-aware routing that maximizes the lifetime of sensor networks. We also identify comparative elements that help us to perform a thorough a posteriori comparison of the mapping functions in terms of the route selection precision. Simulation results show that our mapping functions select routes with very high precision while keeping the propagation delay of routing messages reasonable. In addition, we show that redundant messages filtering significantly extends lifetime of sensor networks compared to classical maximum lifetime approaches in which no filtering is applied.

Keywords: Multihop networks; Sensor networks; Energy-efficient protocols; Routing protocols; Maximum lifetime routing

1. Introduction

Sensor networks are composed of wireless nodes that sense various environmental phenomena and maintain communication interconnection via multihop routing. These easily deployable, self-organized, and relatively low-cost networks are expected to be massively deployed in many applications such as habitat monitoring, disaster relief, and surveillance [1–3]. The success of the applications relies on the network lifetime that depends on the life span of nodes. Hence, energy saving is the crucial factor in designing long-lived sensor networks, mainly because nodes are powered by batteries that may be costly, difficult, or even impossible to replace or recharge.

Designing a universal scheme for optimizing energy savings is challenging due to the variety of sensor network applications. However, for most of applications, measurements presented in the literature [4,5] and obtained from our experiments (Table 1)[1] show that radio communication is a major source of energy consumption. Therefore, many protocols at different layers have been proposed to address this issue [8]. In the rest of this paper, we focus on energy-efficient routing protocols [9].

At the routing layer, energy-efficient protocols use one or a combination of the following strategies to maximize network lifetime: (a) min energy metric and (b) max–min residual energy metric. In min energy routing, nodes select the route that consumes the least amount of energy. Usually, nodes adjust their transmission power and construct a minimum energy topology to reduce the overall energy

* Corresponding author. Address: France Telecom R&D, Meylan, France. Tel.: +33 476 827 228; fax: +33 476 827 287.
E-mail addresses: abdelmalik.bachir@imag.fr (A. Bachir), dominique.barthel@orange-ftgroup.com (D. Barthel), martin.heusse@imag.fr (M. Heusse), andrzej.duda@imag.fr (A. Duda).

[1] We carried out these measurements on the MC 13192 SARD sensor node. The measurements closely match the values announced in datasheets [6,7].

0140-3664/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2007.05.054

Table 1
Current consumption measurements for freescale MC 13192 SARD

Radio idle (not ready to receive)	0.5 mA
Radio Tx (transmit)	39 mA (at +4 dbm)
Radio Rx (receive)	39 mA
MCU (active)	10 mA
MCU (partially active)	8 mA
LED	4 mA
Accelerometer sensors	3 mA

consumption of the network [10,11]. The resulting topology guarantees that each node communicates with other nodes using the route that consumes the least amount of energy possible overall. In max–min residual energy routing, nodes estimate their residual energy and cooperate to prevent the most vulnerable ones from being overused avoiding in this way premature energy exhaustion [12]. Such protocols choose routes bypassing vulnerable nodes, which ensures load balancing and avoids early network fragmentation.

Many research results (see also Section 8) conclude that an energy-efficient routing protocol that maximizes the life span of a sensor network should combine both min energy and max–min residual energy metrics, because these two approaches are complementary. Indeed, at the beginning of the network life, the network is dense and nodes have high residual energy so the use of a pure max–min metric may be counter effective – by trying to protect nodes with low residual energy, the max–min metric always selects routes for which the most vulnerable node has the highest residual energy; such a route may actually dissipate more energy than others. So, the min energy metric, which selects the route with the least energy consumption, is a better choice when nodes have enough energy, i.e. their residual energies are larger than a predefined threshold. The max–min residual energy metric should be used to protect nodes with low residual energy, i.e. less than a predefined threshold.

Although, such hybrid protocols contribute to better network lifetimes, they still have some drawbacks. In this work, we identify the problem of superfluous routing messages that a node may receive while making the best routing decision. Indeed, in traditional routing protocols with metrics such as min energy or max–min residual energy, a node needs to receive routing messages from all of its neighbors to be able to select the best route. This is because these routing messages contain values required for route selection. We argue that the reception and processing (comparison) of all the messages are not needed as the node eventually selects only one route. To address this issue, we propose an approach that enables the best route selection based on exactly one message reception. We call such an approach $O(1)$-reception routing. The key idea of the $O(1)$-reception routing is based on delaying the forwarding of routing messages (RREQ) for a time interval inversely proportional to the residual energy of nodes. This intentionally added delay impacts the propagation of the routing message so that the message coming from the best next hop forwarding candidate is received the first. Afterwards, the node ignore other redundant routing messages that may arrive later on. The performance of $O(1)$-reception routing strongly depends on the choice of the mapping function between the residual energy and the intentional delay.

The rest of the paper is organized as follows. We first present the principles of the proposed routing scheme (Section 2) and formulate the problem of choosing an adequate mapping function (Section 3). Then we consider two approaches, one based on heuristic functions (Section 4) and second one providing an exact solution (Section 5). We evaluate the proposed solutions analytically (Section 6) and through simulation (Section 7). Finally, we discuss some related work (Section 8) and conclude (Section 9).

2. Overview of $O(1)$-reception routing

The $O(1)$-reception routing is based on our energy-delay mapping technique. Therefore, it can be used to enhance any min-delay routing scheme including directed diffusion [13]. Directed diffusion is destination-initiated in the sense that data collectors (also called sinks) query data publishers (also called sources) asking for specific data types. This phase, similar to a route request in on-demand ad hoc routing protocols, is called interest propagation. It establishes localized data-forwarding pointers (called gradients) from sources to sinks. Sources then stream the requested data back to sinks according to the directions indicated by the gradients. Although, there are different implementations of gradient routing, one phase pull directed diffusion is the best fit when few sinks collect the data published by many sources [14]. Since such situations are fairly common in sensor network applications, we only consider one phase pull directed diffusion[2] in this paper.

Our motivations for using diffusion are the following:

- Computational complexity is reduced to a minimum. Each node only needs to broadcast one interest message during the interest propagation phase and it only needs to receive one interest message to setup its routing table (it can ignore the subsequent interest messages related to that same interest). Note that this situation is beneficial only if the underlying MAC protocol enables filtering redundant messages. In a previous work [15,16], we have shown how to enable this feature in both preamble sampling and common active/sleep schedules based MAC protocols. Filtering redundant messages allows a node to switch its radio-off during redundant receptions, which saves energy.
- There is no overhead due to the exchange of extra information like hello or route metrics messages, which saves

[2] Which we will simply call diffusion.

more energy and reduces the computation complexity and memory occupation of the routing protocol.

- Routing tables only require one entry per active interest consisting of a pointer toward the next node downstream.
- It enables in-network processing to aggregate data based on attributes used in diffusion, which saves more energy by reducing the amount of transmitted and received messages.

The $O(1)$-reception routing enhances the basic diffusion routing scheme by delaying the interests forwarding for an interval inversely proportional to the residual energy: nodes compute a forwarding delay based on their residual energy and defer the forwarding of interest messages for this period of time. As maximum lifetime routing should combine the min and the max–min metrics, the energy-delay mapping function should have the following properties: nodes with high residual-energy forward interests without delay to make diffusion equivalent to the min energy routing, and nodes with low residual-energy delay forwarding of interests for a time interval to make diffusion equivalent to the max–min residual energy routing.

In Fig. 1, we present the principles of such an energy-delay mapping technique. In Fig. 1a, all nodes have high residual energies thus they do not add any intentional delay. Therefore, the selected route is the shortest one as the interests propagation on which is the fastest. In our case, this corresponds to the min energy routing as the shortest route is the minimum energy consumption route, because we consider that all links have the same energy consumption. In Fig. 1b, we illustrate the max–min part of the algorithm that is used to protect nodes with low residual energies. The overused node shown in Fig. 1b, which is on the shortest route, should choose its intentional forwarding delay so that interest propagation on the other route is faster and thus the route is selected according to the max–min metric.

3. Problem statement and system model

The main problem in our routing scheme is energy-delay mapping, i.e. how to relate the residual energy to the intentional delay. Turning a min-delay metric into the min energy metric is fairly straightforward when all links have equal energy consumption: it is sufficient to add no extra intentional delay. However, turning a min-delay metric into the max–min residual-energy metric is much more complex. Therefore, we formulate the problem as follows.

Assume $\mathcal{R}$ is the set of all possible routes between a sink node and a source node. We call $|R_k|$the number of intermediate nodes on route R_k ($R_k \in \mathcal{R}$), source and destination nodes are not included. We use the following notation to represent R_k, $R_k = N_{1k} - \cdots - N_{ik} - \cdots - N_{|R_k|k}$, where N_{ik} represents an intermediate node on route R_k.

We assume that each node is able to measure its residual energy and we call ζ_{ik} the relative residual energy of node N_{ik}. Values ζ_{ik} are normalized in $[0,1]$, i.e. $0 \leqslant \zeta_{ik} \leqslant 1$ for all nodes.

We call ζ_k^- the node with the least amount of residual energy on route R_k. We have:

$$\zeta_k^- = \min_{1 \leqslant i \leqslant |R_k|} \{\zeta_{ik}\}. \tag{1}$$

The max–min residual energy routing selects the route with the largest ζ_k^-, i.e. it selects the route R that satisfies:

$$R = \underset{R_k \in \mathcal{R}}{\operatorname{argmax}} \{\zeta_k^-\}. \tag{2}$$

By combining Eqs. (1) and (2), we obtain:

$$R = \underset{R_k \in \mathcal{R}}{\operatorname{argmax}} \left\{ \min_{1 \leqslant i \leqslant |R_k|} \{\zeta_{ik}\} \right\}. \tag{3}$$

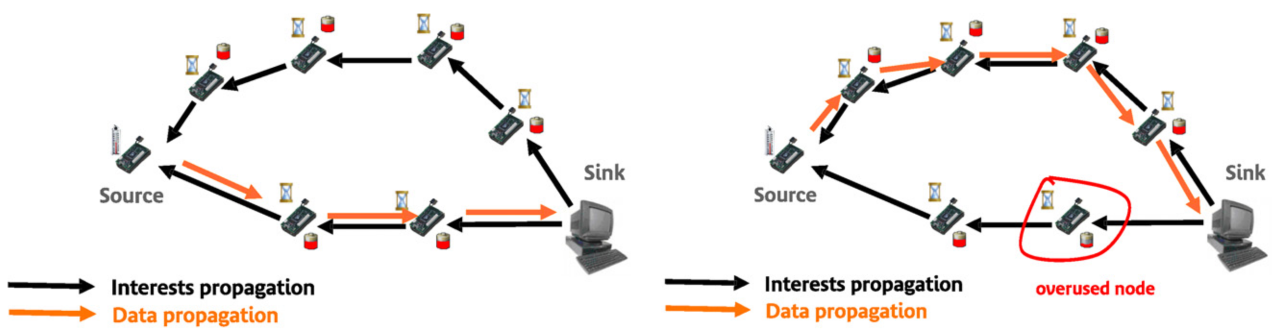
Let us now examine min-delay routing. We call D_{ik} the delay introduced by each node N_{ik} on route R_k. Route R_k experiences the total delay of $D(R_k)$. We have:

$$D(R_k) = \sum_{i=1}^{|R_k|} D_{ik}. \tag{4}$$

The min-delay routing selects the route with minimum delay. Therefore, the selected route, denoted by R', satisfies:

$$R' = \underset{R_k \in \mathcal{R}}{\operatorname{argmin}} \{D(R_k)\}. \tag{5}$$

By combining Eqs. (4) and (5), we obtain:



(a) Turning diffusion into min energy routing.

(b) Turning diffusion into max-min residual energy routing.

Fig. 1. Principles of energy-delay mapping technique.

$$R' = \underset{R_k \in \mathcal{R}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{|R_k|} D_{ik} \right\}. \quad (6)$$

Our goal is to make the min-delay routing select the route that satisfies the max–min residual energy metric, i.e. make route R' match route R. The next sections show how we achieve this goal by two different means: heuristics and exact solutions.

4. Approximate solution: heuristic functions

To make route R' match route R, we propose to use a function f to map the residual energy of nodes into an intentional delay. Our goal is to solve Eq. (3) by solving Eq. (6) on a suitable set of:

$$D_{ik} = f(\zeta_{ik}). \quad (7)$$

By choosing f to be strictly decreasing, we can rewrite Eq. (3) as:

$$R = \underset{R_k \in \mathcal{R}}{\operatorname{argmin}} \left\{ f\left(\min_{1 \leqslant i \leqslant |R_k|} \{\zeta_{ik}\} \right) \right\}. \quad (8)$$

By matching Eq. (8) with Eq. (6) and replacing D_{ik} by its values calculated in Eq. (7), we conclude that function f that meet our goal should satisfy the following equation (Eq. (9)) for all i in $1, \cdots, |R_k|$:

$$\sum_{i=1}^{|R_k|} f(\zeta_{ik}) = f\left(\min_{1 \leqslant i \leqslant |R_k|} \{\zeta_{ik}\} \right). \quad (9)$$

We can obtain an approximate solution by choosing f to be convex and decreasing in $[0,1] \to [0,1]$ so that the minimal ζ_k^- along route R_k makes a dominant contribution to the sum to the left of Eq. (9), i.e. we have:

$$f(\zeta_k^-) \gg \left(\sum_{i=1}^{|R_k|} f(\zeta_{ik}) - f(\zeta_k^-) \right). \quad (10)$$

Therefore

$$\sum_{i=1}^{|R_k|} f(\zeta_{ik}) \approx f\left(\min_{1 \leqslant i \leqslant |R_k|} \{\zeta_{ik}\} \right). \quad (11)$$

which is an approximate solution for Eq. (9).

To find a mapping function f with suitable properties, we have explored a family of decreasing convex functions of the form $(1/x)^\eta$, where η is a positive parameter. We have shifted and shrunk them so that they map the residual energy in $[0,1]$ into the normalized delay in $[0,1]$. In Fig. 2, we present the resulting set of functions labeled f_η with η taking integer values from 1 to 4.

We use the parameter η to control the convexity of the mapping function that determines its ability to approximate max–min routing. The purpose of the parameter is to influence the intentional delay applied by the node with the minimum residual energy on a route so that it will be dominant. In this way, the route with the max–min residual energy will be selected, because the interest propagation on this route will have minimal delay. The convexity determines the precision of the approximation in Eq. (11): the more convex the mapping function, the better the approximation. For example, function f_4 has stronger convexity than the other functions in the considered set. Therefore, it approximates max–min routing better. However, functions with very high convexity such as f_4 present an inherent drawback resulting from their weak sensitivity threshold. The sensitivity threshold is the value that separates the flat part of the function from the curvy one. For example, function f_3 has a sensitivity threshold of 0.5, which means that a node using this mapping function will not apply any intentional delay when its residual energy is larger than 0.5. Therefore, if we have routes with nodes having residual energies larger than 0.5, the selected route will be the one with the min delay, which very likely corresponds to the shortest path consuming the minimum energy. We can say that function f_3 uses a maximum lifetime routing with a battery protection threshold of 0.5. The battery protection threshold differentiates low residual-energy nodes from high residual-energy ones.

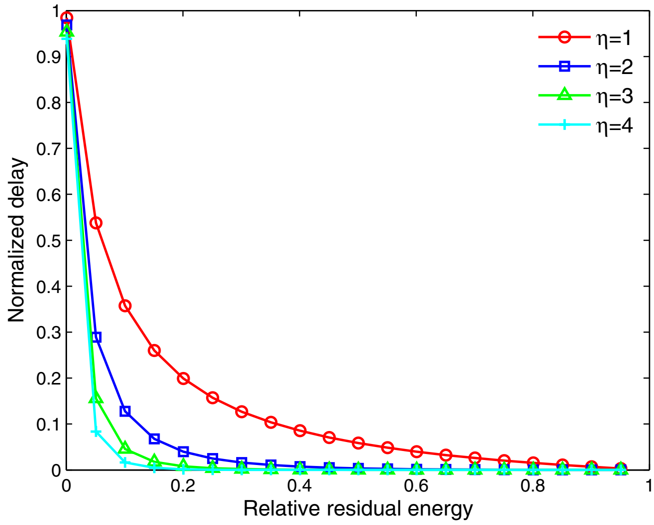


Fig. 2. Heuristic mapping functions.

Note that there is a relation between the convexity of the function and the sensitivity threshold. More accurate max–min routing requires higher convexity functions (e.g. f_4), which results in smaller battery protection thresholds as higher convexity functions have smaller sensitivity thresholds. Vice versa, larger battery protection thresholds imply using larger sensitivity-threshold functions (e.g. f_1), which results in less max–min precision.

To overcome this shortcoming, we propose in the next section a synthetic mapping function that performs an exact transformation of the min metric into the max–min one according to an uncorrelated predefined battery protection threshold. This mapping function is to be used in the situation in which the residual energies of nodes are expressed as step functions and not continuous ones.

5. Exact solution: synthetic function

In real implementations of routing protocols, the energy-delay mapping function would likely be discrete and tabulated. A node may read its battery voltage or integrate the consumed current and perform a table lookup to get the corresponding level of its residual energy. Therefore, we can assume that the residual energy of nodes is discrete.

We call γ the battery protection threshold, $(0 < \gamma < 1)$. Then, a node is *vulnerable*, if its residual energy is less than battery protection threshold γ. A node is *critical* for a route, if it has the least amount of residual energy among all the nodes forming that route. The *residual energy of a route* is equal to the residual energy of the critical node for that route. A *route is vulnerable*, if its residual energy is less than γ.

We aggregate all the energy levels greater than γ into one energy level as shown in Fig. 3. We quantize the energy level below γ in m even levels: a discrete energy level l corresponds to the residual energy ζ if:

$$(l-1)\frac{\gamma}{m} < \zeta \leqslant l\frac{\gamma}{m}. \tag{12}$$

If ζ is larger than γ, the node is assigned discrete energy level $m+1$. Therefore, we have:

$$l = \begin{cases} \left\lceil \frac{m\zeta}{\gamma} \right\rceil & \text{if } \zeta \leqslant \gamma \\ m+1 & \text{otherwise.} \end{cases} \tag{13}$$

Let g be a synthetic function that maps residual energy into intentional forwarding delay d: $d = g(\zeta)$. As we use discrete energy levels instead of continuous residual energy, function g is dependent on m. Therefore, the intentional forwarding delay $d(l)$ that corresponds to energy level l is the following:

$$d(l) = g_m(l). \tag{14}$$

The synthetic function g that meets our goals even in the worst case needs to be decreasing so that $g(l) < g(l')$ for all $l > l'$. In addition, g also needs to be convex to mitigate the effect of increasing delay cumulated along longer routes. In Fig. 4, we shows the worst case with an example with two routes R_k and $R_{k'}$. Route R_k has the maximum route length $|R_k| = n$ and a residual energy level of l. However, route $R_{k'}$ has the minimum route length $|R_{k'}| = 1$ and a residual energy level of $l-1$. To select the best route (route R_k), the interest propagation delay $D(R_k)$ on route R_k should be less than $D(R_{k'})$. As we assume these delays to be discrete, it is sufficient to have:

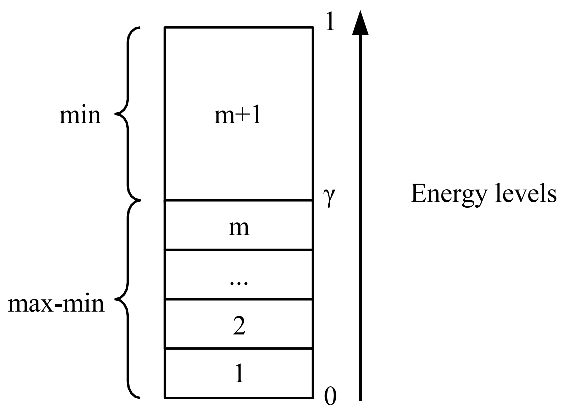


Fig. 3. Energy levels.

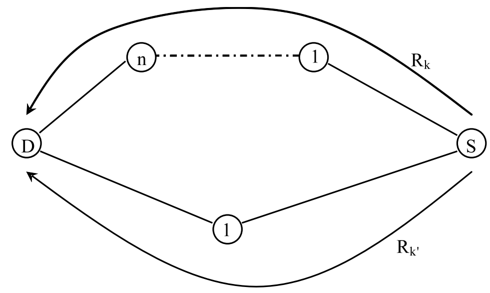


Fig. 4. The worst case illustrated with two routes.

$$D(R_{k'}) = D(R_k) + 1. \tag{15}$$

Therefore,

$$\sum_{i=1}^{|R_{k'}|} D_{ik'} = \sum_{i=1}^{|R_k|} D_{ik} + 1, \tag{16}$$

where D_{ik} is the delay incurred by node N_{ik}. Actually, the delay D_{ik} is composed of two parts: an intentional delay d_{ik} controlled via the synthetic mapping function and an inherent system delay δ_{ik} that includes computation and transmission delays. For example, in contention-based medium access protocols, the system delay includes the average backoff time used to reduce collision rates. Thus, we have:

$$D_{ik} = d_{ik} + \delta_{ik}. \tag{17}$$

In the worst case, nodes on route R_k experience maximum system delays, i.e. $\delta_{ik} = \delta_{\max}$ and nodes on route $R_{k'}$ experience minimum system delay $\delta_{ik'} = 0$. Moreover, all nodes on route R_k have their energy levels equal to l, i.e. $d_{ik} = g_m(l)$ for $i = 1, \cdots, |R_k|$and the node on route $R_{k'}$ has its energy level equal to $l-1$, i.e. $d_{ik'} = g_m(l-1)$ for $i = 1, \cdots, |R_{k'}|$. Therefore, Eq. (16) can be rewritten as:

$$g_m(l-1) = n[g_m(l) + \delta_{\max}] + 1. \tag{18}$$

We set $g_m(m+1)$ to 0 so that non-vulnerable nodes do not apply any intentional delay, which performs min energy routing without any added delay. Therefore, Eq. (18) rewrites to as:

$$g_m(l) = \begin{cases} (n\delta_{\max} + 1)\frac{n^{m-l+1}-1}{n-1} & \text{if } l \leqslant m \\ 0 & \text{otherwise.} \end{cases} \tag{19}$$

6. Analytical evaluation

As the synthetic function is more suitable for real implementations than the heuristic and theoretical function, we evaluate in this section the end-to-end intentional added delay for these functions. We also analyze the rate of vulnerable routes as the intentional delay applies only to vulnerable routes.

6.1. Worst case interest propagation delay

Assume that there are n intermediate nodes $N_1, \ldots, N_n$ between the source and the destination. Each node N_i has residual energy level l_i. On route $R = N_1 - \ldots - N_n$, node N_i receives the interest at time t_i (we assume the destination sends the interest at time 0):

$$\begin{cases} t_1 = \delta_1 \\ t_2 = (g(l_1) + \delta_2) + \delta_1 \\ t_3 = (g(l_2) + \delta_3) + (g(l_1) + \delta_2) + \delta_1 \\ \vdots \\ t_{n+1} = \sum_{i=1}^{n} (g(l_i) + \delta_{i+1}) + \delta_1, \end{cases} \tag{20}$$

where t_{n+1} is the time when the source receives the interest.

In the worst case, all intermediate nodes N_i, $i = 1, \ldots, n$ have residual energy levels of 1 (i.e. $l_i = 1$ for all $i = 1, \ldots, n$) and all system delays $\delta_i = \delta_{\max}$ for all $i = 1, \ldots, n$. Hence, the maximum interest propagation delay in the worst case corresponds to the maximum value of t_{n+1}, which is:

$$\begin{aligned} D_{\max} &= n\left(n^{m-1} - 1\right)\left(\delta_{\max} + \frac{1}{n-1}\right) \\ &= O(n^m \delta_{\max}). \end{aligned} \tag{21}$$

6.2. Vulnerable routes rate

We propose to analyze the probability with which a node uses min or max–min metrics to select routes. This probability depends on parameters shown in Table 2.

A node picks out a route according to the max–min metric if all the routes are vulnerable. Then,

Table 2
Notation

p_γ	Probability that a node is not vulnerable
$\lvert\mathcal{R}\rvert$	Number of disjoint routes between the source and the sink
$\lvert R_k\rvert$	Length of route R_k
n	Number of intermediate nodes on the longest route between the source and the destination
$P_{\min}(k)$	Probability that route R_k is not vulnerable
$P_{\text{maxmin}}(k)$	Probability that route R_k is vulnerable
P_{maxmin}	Probability that the node selects a vulnerable route

$$\begin{aligned} P_{\text{maxmin}} &= \prod_{k=1}^{|\mathcal{R}|} P_{\text{maxmin}}(R_k) \\ &= \prod_{k=1}^{|\mathcal{R}|} (1 - P_{\min}(R_k)). \end{aligned} \tag{22}$$

A route is not vulnerable if all the intermediate nodes on that route are not vulnerable. Therefore,

$$P_{\min}(R_k) = \prod_{i=1}^{|R_k|} p_\gamma, \tag{23}$$

and,

$$P_{\text{maxmin}} = \prod_{k=1}^{|\mathcal{R}|} \left(1 - \prod_{i=1}^{|R_k|} p_\gamma\right). \tag{24}$$

The mean $E[P_{\text{maxmin}}]$ is the following:

$$E[P_{\text{maxmin}}] = \left(E[1 - p_\gamma^L]\right)^{|\mathcal{R}|}, \tag{25}$$

where L is a random variable that expresses route lengths. We have:

$$E[1 - p_\gamma^L] = \sum_{i=1}^{n} \left(1 - p_\gamma^i\right) \cdot P\{L = i\}. \tag{26}$$

We assume L being a discrete uniform random variable in $[1, n]$, i.e. $P\{L = i\} = 1/n$. Thus

$$E[1 - p_\gamma^L] = \frac{1}{n}\left(n - \sum_{i=1}^{n} p_\gamma^i\right). \tag{27}$$

Finally,

$$E[P_{\text{maxmin}}] = \left[1 - \frac{p_\gamma}{n}\left(\frac{1 - p_\gamma^n}{1 - p_\gamma}\right)\right]^{|\mathcal{R}|}. \tag{28}$$

From Eq. (28) and Fig. 5, we conclude that the probability of selecting a route according to max–min (i.e. all the routes are vulnerable) decreases when the number of routes $|\mathcal{R}|$ increases. This means that in dense networks in which there are many alternative routes, finding a not vulnerable route becomes very likely. We also notice that probability P_{maxmin} increases when the number of intermediate nodes n increases, which is quite expected. Besides, when probability p_γ that a node is not vulnerable increases, probability P_{maxmin} that all the routes are vulnerable decreases, because the number of vulnerable nodes decreases.

7. Simulations

We have used ns-2 [17] to evaluate our synthetic mapping functions when used with diffusion. The goal of these experiments is to observe the lifetime extension obtained through the use of our mapping functions, the corresponding end-to-end interest propagation delay, and the benefits of filtering redundant interests. As the lifetime extension also depends on the energy-efficiency of the MAC protocol

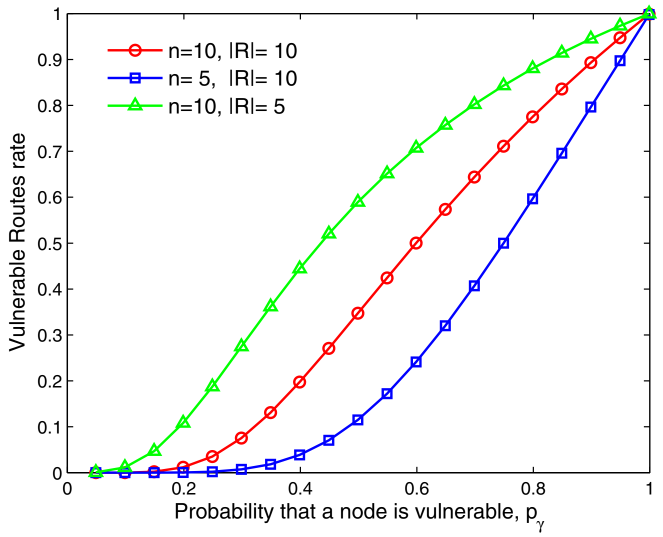


Fig. 5. The rate of vulnerable routes in function of the probability that a node is vulnerable. $|\mathcal{R}|$ is the number of disjoint routes between the source and the destination.

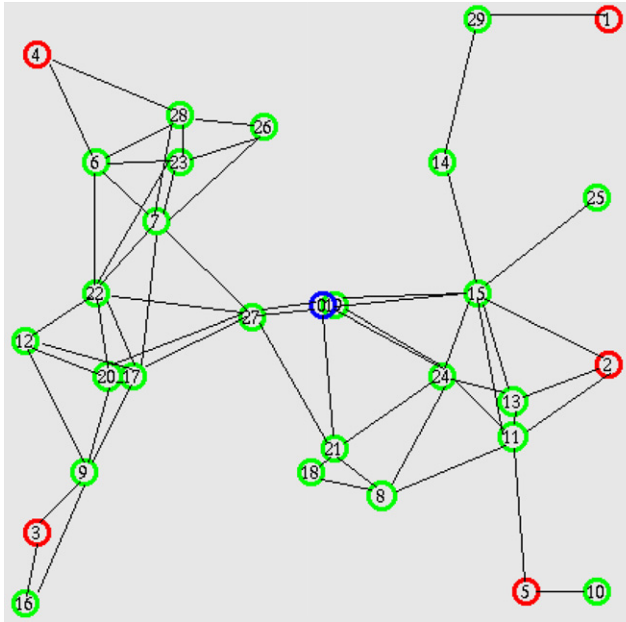


Fig. 6. Random network.

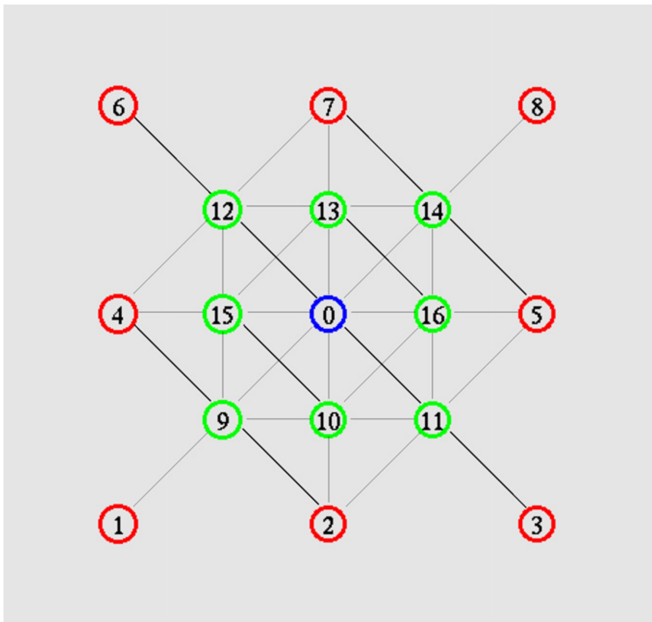


Fig. 7. Star network.

beneath, we have run simulations with two types of MAC protocols: an ideal MAC and MFP [15]. The ideal MAC has no idle listening (i.e. a node consumes energy only when it transmits or receives a message) and no collisions. Therefore, it allows us to quantify the benefit of the mapping function independently of the MAC protocol performance. The MFP protocol allows us to show the expected performance with a real MAC protocol. To evaluate the performance of filtering redundant interests, we use an ideal MAC that filters out redundant messages before their receptions. We call this protocol Ideal-filter to distinguish it from Ideal-nofilter that does not filter out redundant messages. Likewise, we activate the filtering option of MFP in MFP-filter and we distinguish if from MFP-nofilter.

We have carried out experiments on two networks topologies: a random topology (Fig. 6) and a star topology (Fig. 7). In the former, the sink is Node 0 and the sources are Nodes 1 through 5. In the latter, the sink is Node 0 and the sources are Nodes 1 through 8. The sink generates interests every 100 s for refreshing existing routes or finding new ones. Every source that receives an interest sends data back to the sink according to the gradient installed by the interests. Each source sends a data message every 30 s. We have used a simple energy model in which transmission and reception powers are equal. The sink have unlimited initial energy, whereas the other nodes including sources have enough initial energy so that a significant amount of data messages are received by the sink from each source.

For the experiments, we consider two performance parameters: the lifetime extension achieved by the $O(1)$-reception routing protocol compared to the basic diffusion protocol for each source and the end-to-end interest propagation delay from the sink to sources. We calculate the lifetime of each source in function of the number of transmitted data messages that successfully reach the sink before the source loses connection with the sink.

In the first experiments, we have set the number of energy levels m to 4 and varied the battery protection threshold γ from 0.1 to 0.9. We have measured the resulting lifetime extensions obtained in the best case for both topologies, i.e. when an Ideal-filter MAC protocol is used. In Fig. 8, we plot two measures of the lifetime extension: the avg-lifetime and the max-lifetime. The avg-lifetime is obtained by averaging out all the lifetime extensions by all the sources, and the max-lifetime is the lifetime exten-

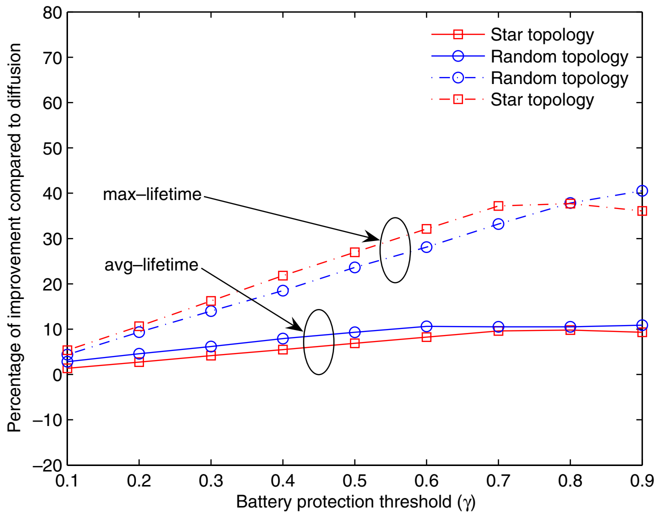


Fig. 8. Lifetime extension according to γ with the Ideal-filter MAC protocol. The value of m is set to 4.

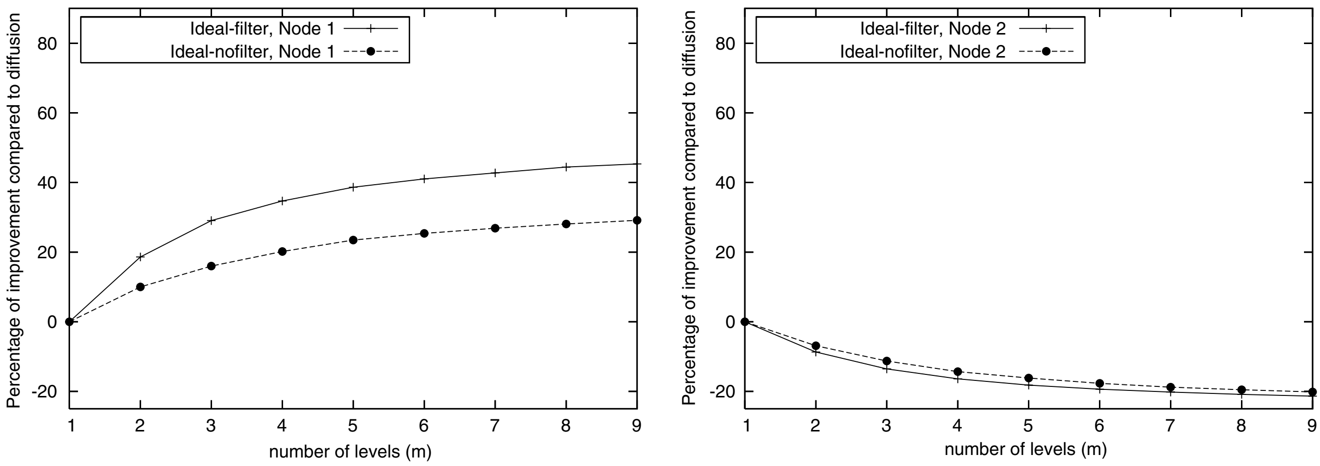
sion of the source that obtained the maximum lifetime extension. In Fig. 8, we show that the random and the star topologies have the same behavior: the avg-lifetime and the max-lifetime increase when the battery protection threshold increases for both topologies. Therefore, we conclude that a large battery threshold is better for these situations. For the next experiments, we set the battery protection threshold γ to 1.

For the second experiments, we have varied m from 1 to 9 to evaluate the trade-off between the lifetime extensions and the end-to-end interest propagation delays. In Fig. 9, we show the lifetime extensions obtained with Source 1 (Fig. 9a) and Source 2 (Fig. 9b) in the star topology. We have plotted lifetime extensions only for these sources because there is a symmetry in the star topology: the results obtained for Sources 3, 6, and 8 are the same as those obtained for Source 1, and those obtained for Nodes 4, 5, and 7 are the same as those obtained for Source 2.

As expected, Fig. 9a shows that the lifetime extension increases when the number of levels m increases, because the more energy levels we have, the more accurate our mapping function is. However, the percentage of lifetime extension increases with less intensity. That is, increasing m from 2 to 3 increases the lifetime extension by a factor that is smaller than that when increasing m from 1 to 2.

Note that increasing the lifetime of some sources may decrease that of other sources, which results in some sources with negative lifetime extensions as shown in Fig. 9b. Source 2 (and also Sources 4, 5, and 7) has a negative lifetime extension, because their lifetime with diffusion routing is longer than that with $O(1)$-reception routing. With diffusion, Source 2 has three potential relays (Nodes 9, 10, and 11). However, with $O(1)$-reception routing, Source 2 mostly has only one relay (Node 10), because Nodes 9 and 11 relay the traffic of Sources 1 and 3, respectively. Note that even with these negative lifetime extensions, the overall lifetime extension (the avg-lifetime in Fig. 8) is positive.

We have carried out the same experiments on the random topology and obtained the following results: Sources 1 and 5 have positive lifetime extensions, Sources 3 and 4 have zero lifetime extensions, and Source 2 has negative lifetime extension. Sources 3 and 4 have zero lifetime extensions, because all their traffic passes through Node 27. As there are no alternative routes for these sources, no lifetime extension will be achieved no matter how well the routing algorithm performs. Source 2 has a negative lifetime for the same reasons explained above with the star topology. For Sources 1 and 5, Fig. 10 shows the percentage of their lifetime extensions. We can see that the lifetime extension for Source 1 is larger then that for Source 5. Two key Nodes (15 and 11) that are critical for the lifetime of Sources 1 and 5. respectively, cause this result. With diffusion, Source 1 has a lifetime that is smaller than that of Source 5, because the only route that connects Source 1 with the sink contains Node 15. This route is more vulner-



(a) node 1 (b) node 2

Fig. 9. Percentage of improvement with $\gamma = 1.0$ for a star network.

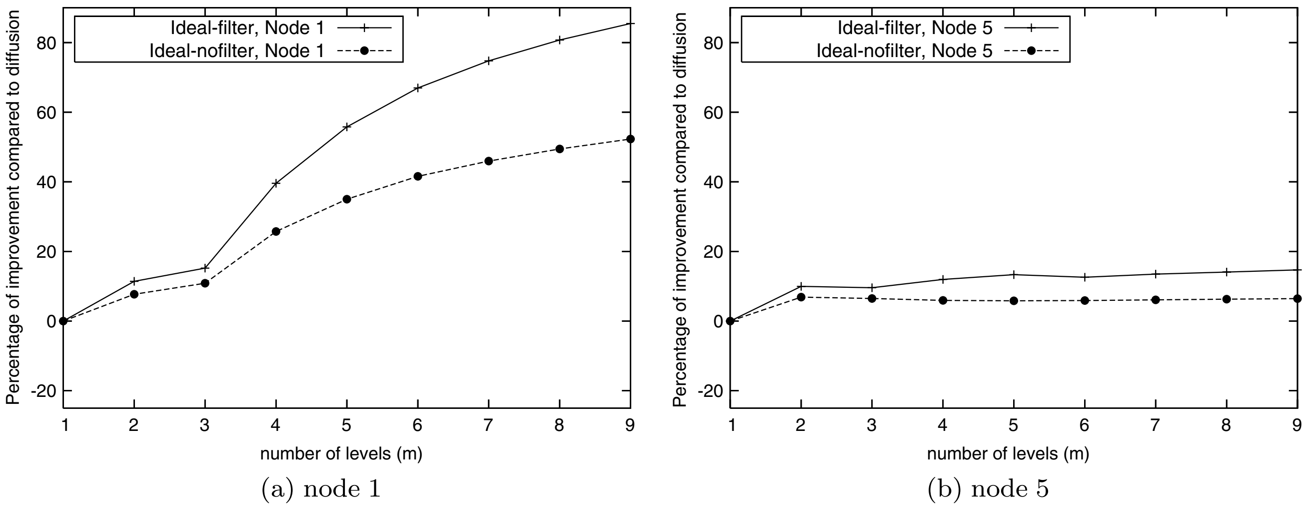


Fig. 10. Percentage of improvement with $\gamma = 1.0$ for a random network.

able than the other routes connecting Source 5 to the sink, because Node 15 relays most of the traffic of Source 2 as it is on the shortest route from Source 2 to the sink, which is route (2–15–0). Moreover, Node 15 is more vulnerable than Node 11, because it receives and sends more interests as it has a higher number of neighbors. As our algorithm protects vulnerable nodes from being overused, the lifetime of Node 15 increases with a percentage that is larger than that of Node 11, thus increasing the lifetime of Sources 1 and 5 accordingly.

From Figs. 9a, 10a and b, we can see that increasing the number of energy levels m increases the lifetime of sources connected to the sink through vulnerable routes. However, it is expected to increases the end-to-end interest propagation delays. Therefore, we need to make a trade-off between lifetime extension and interest propagation delays by choosing a suitable value for m. For this, we present in Figs. 11 and 12 the average end-to-end interest propagation delays experienced in diffusion and in $O(1)$-reception routing for the star and the random topologies, respectively. These figures confirm the derivations carried out in Section 6.1 that show that the end-to-end interest propagation delay increases exponentially when m increases linearly. For example, when $m = 4$, we obtain a substantial lifetime extension with an almost negligible end-to-end interest propagation delay.

In addition to increasing the lifetime of sources by using an energy-efficient metric, $O(1)$-reception routing also increases the lifetime of the network by reducing the overhead of exchanged messages. Fig. 10a shows that avoiding the reception of redundant messages at the MAC layer (Ideal-filter) allows Node 1 to increase its lifetime by up to 40% compared to when no filtering (Ideal-nofilter) is used. This significant lifetime extension percentage is mainly due to the improvements realized by Node 15 that has a large number of neighbors. When there is no filtering, Node 15 receives all the interests forwarded by its neighbors, i.e. 8 interests with the same information. However,

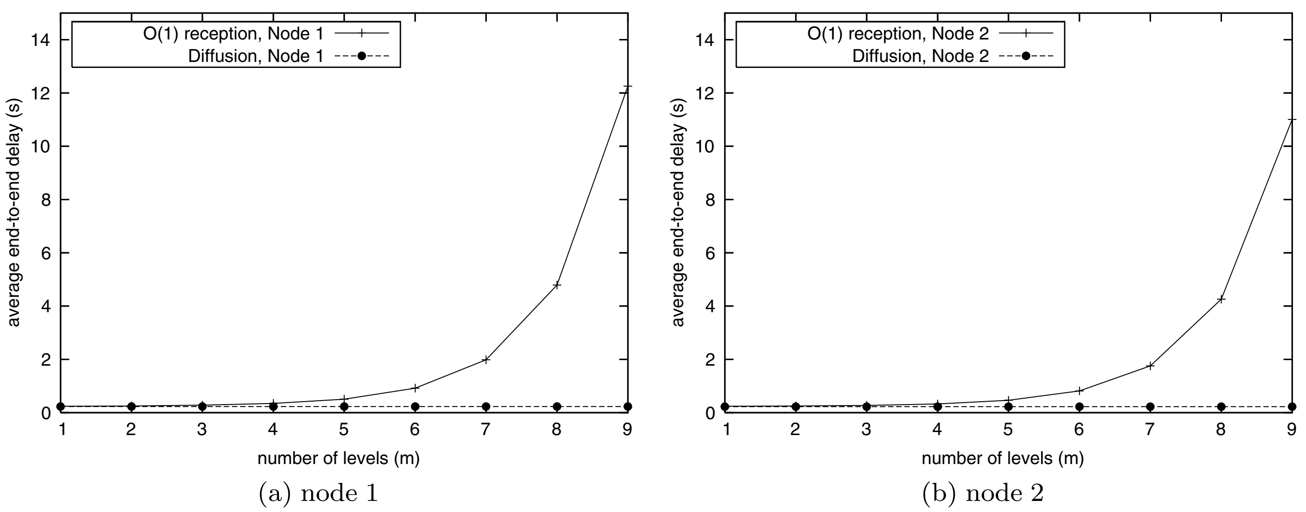
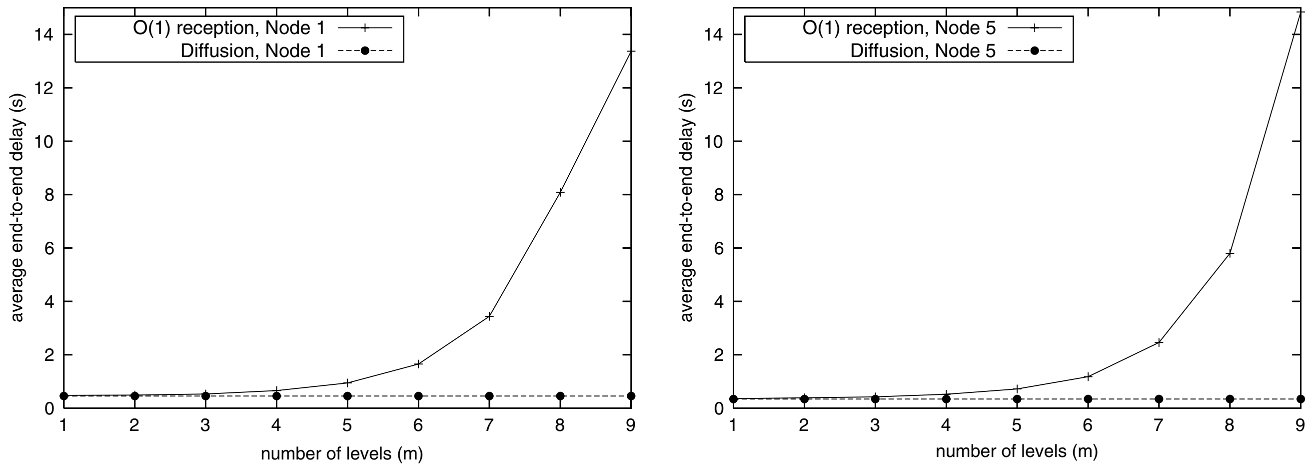


Fig. 11. Average interest propagation delay with $\gamma = 1.0$ for a star network.



(a) node 1 (b) node 5

Fig. 12. Average interest propagation delay with $\gamma = 1.0$ for a random network.

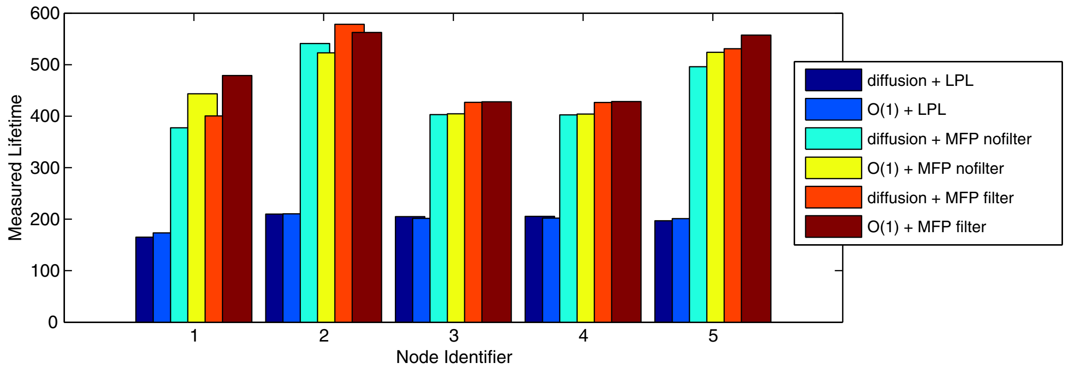


Fig. 13. Lifetime extension for each node in the random topology.

when filtering is used, Node 15 receives only 1 interest as it filters out the redundant interests. We expect that filtering will achieve further energy saving in more dense networks.

In Fig. 13, we plot the results we obtained for $O(1)$-reception routing with realistic MAC protocols: LPL, MFP-filter, and MFP-nofilter. The results obtained for Sources 1 through 5 confirm the arguments presented above. They also show that we obtain a substantial[3] lifetime when jointly using our contributions: MFP-filter and $O(1)$-reception routing.

8. Related work

Toh et al. [18] have proposed CMMBCR (Conditional Max–Min Battery Capacity Routing) for the network lifetime maximization problem. CMMBCR is a combination between MTPR, the min energy metric, and MMBCR, the max–min residual energy metric. In their proposal, they define battery protection margin γ, $(0 \leqslant \gamma \leqslant 100)$ and differentiate two kinds of routes: A and Q. Q is the set of all possible routes between a source and a destination nodes. A, a subset of Q, is the set of the routes having residual energy greater than γ, i.e. all the nodes on each route in A have residual energies larger than γ. The protocol is the following: when there is no route in A with residual energy below γ (i.e. all the possible routes contain vulnerable nodes), the protocol selects a route in Q according to the max–min residual energy routing (MMBCR) to protect the most vulnerable nodes. Otherwise, when there is at least one route in A, the algorithm selects a route in A according to the min energy routing (MTPR) to save energy. Note that γ is the parameter that controls the trade-off between MMBCR and MTPR.

Misra and Banerjee [19] take the link transmission cost between nodes into account and propose MRPC (Maximum Residual Packet Capacity) to improve the previous protocol. They model the link transmission cost according to the link error rate and the physical distance between nodes. They introduce a node-link metric C_{ij}, for each link $i \rightarrow j$, that depends on the residual energy B_i of Node i, and on the transmission power ζ_{ij} needed to send a packet from i to j. Explicitly, $C_{ij} = B_i/Eij$. The node-link metric determines the lifetime of the link $i \rightarrow j$. The lifetime Life_R of route R depends on the lifetime of the most vulnerable link

[3] Note that as explained in the previous section, the $O(1)$-reception routing marginally reduces the lifetime of Node 2 when it aims at increasing the lifetime of Node 1 and Node 5.

on this route, $\text{Life}_R = \min\{C_{ij}\}$, where $i \to j$ is a link on route R. The protocol is then straightforward: given a set of routes between a source and a destination node, choose the route with the largest lifetime. Note that basic MRPC is a pure max–min residual energy routing, which could have undesirable behavior by always tending to protect the most vulnerable link. To cope with this issue, Misra and Banerjee, propose CMRPC (Conditional MRPC) that uses life protection threshold γ by analogy to the battery protection threshold [18]. That is, CMRPC first tries to select the route with the minimum energy consumption among the routes whose lifetimes are larger than γ. Otherwise, if there is no route satisfying this condition, CMRPC switches to MRPC. Simulation results show that CMRPC improves the performance of MPCR, in terms of lifetime maximization only if the control parameter γ is well determined.

Li et al. [20] address the network lifetime maximization problem with max–min $zP_{\min}$, an on-line message routing protocol. It first computes $P_{\min}$, the minimum energy needed to transmit a packet from a source node to a destination node across all possible routes. It then uses max–min residual energy metric to pick a route, thereby balancing the load among different nodes, unless the cost is higher than $zP_{\min}$, ($z \geqslant 1$), in which case, it falls back to the min metric thus avoiding excessive energy consumption. The authors propose a centralized algorithm based on the gradient descent technique to determine the optimal value of z. Further on the same authors describe a distributed version of the algorithm [21], but it requires establishing synchronized mini slots at the MAC layer.

Shah and Rabaey [22] consider the drawbacks of pure minimum energy routing for the survivability of the network. They propose a probabilistic route selection scheme to relieve workload of minimum energy routes. Their protocol is the following: given a set of routes between a source and a destination node, assign to each route the probability of being selected so that the minimum energy route has the highest probability. Then, forward packets on routes according to their probabilities. Note that routes with too much energy consumption, by analogy to the max–min $zP_{\min}$ algorithm [20], are assigned zero probability and will never be selected. However, this protocol requires to explicitly transmit link cost information and to receive packets from all routes in order to compute the corresponding selection probabilities.

The discussed papers [22,21,19,18] emphasize the idea of combining the minimum energy and max–min residual energy metrics to optimize the lifetime of sensor networks. However, the distributed nature of these protocols requires explicit transmission of the energy information which is counter productive with respect to energy optimization. Taking this overhead into account and inspired by other papers [23,24], Guo [25] proposes a lightweight broadcast scheme for network lifetime maximization. His protocol encourages nodes with high residual energy to retransmit a broadcast message and works as follows. When a node receives a broadcast message, it delays the retransmission of this message to see if there is another node with higher residual energy. This delay is inversely proportional to the residual energy of the node. Guo's algorithm reduces the number of nodes forwarding a broadcast message without the overhead of explicitly exchanging the residual energy information, but it may miss some nodes in a sparse network.

9. Conclusion

Maximizing the lifetime of a sensor network requires an energy-efficient routing protocol on top of an energy-efficient MAC protocol. In this paper, we have tackled the problem of selecting energy-efficient routes while reducing the overhead of routing protocols. We have proposed a technique called $O(1)$-reception that enables the best route selection based on exactly one routing message reception, thus allowing substantial overhead reduction because in traditional routing a node needs to receive routing messages from all of its neighbors to be able to select the best route.

The $O(1)$-reception routing is suitable for WSNs not only because it reduces reception overhead but also because it can be used with any metric that can be mapped on top of the min-delay metric. In this paper, we have proposed an example in which $O(1)$-reception is used to perform a hybrid min and max–min routing with directed diffusion, allowing thus to benefit from the advantages of a data-centric communication scheme such as traffic aggregation.

The key idea of the $O(1)$-reception routing is based on delaying the forwarding of routing messages for a time interval that is inversely proportional to the residual energy of nodes. This intentionally added delay influences the propagation of routing messages so that the first received one indicates the best route and thus all the subsequent routing messages with the same content can be ignored.

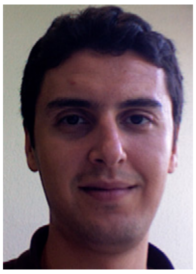
The intentional mapping delay is calculated according to a mapping function that determines the corresponding delay in function of the residual energy of a node. We have shown how to find such functions when the residual energy is continuous and when it is a discrete measure. As in practical implementations the residual energy is discrete, we have analytically evaluated the end-to-end intentional delay in the worst case and the percentage of vulnerable routes in which this delay is added.

We have run extensive simulations with ns-2 to evaluate the performance of $O(1)$-reception routing. As the performance of energy-efficient routing also depends on the MAC protocol beneath, we have considered two MAC protocols: an ideal MAC protocol to only evaluate the benefit of our routing protocol and a real MAC protocol (MFP) to evaluate their combined benefit. The obtained results show that using MFP jointly with $O(1)$-reception routing achieves a substantial lifetime extension.

References

[1] R. Szewczyk et al., Habitat monitoring with sensor networks, Communications of the ACM 47 (6) (2004) 34–40.

[2] E. Cayirci, T. Cuplu, SENDROM: sensor networks for disaster relief operations management, in: Proceedings of MedHocNet, Bodrum, Turkey, June 2004.
[3] T. He et al., Energy-efficient surveillance systems using wireless sensor networks, in: Proceeding of the ACM MobiSys, Boston, MA, June 2004.
[4] Ember Corporation, EM250 Single-Chip ZigBee/802.15.4 Solution, Data Sheet, 2005.
[5] J. Polastre, R. Szewczyk, D. Culler, Telos: enabling ultra-low power wireless research, in: Proceedings of IPSN/SPOTS, Los Angeles, CA, April 2005, pp. 302–311.
[6] Freescale Semiconductor, MC13192/MC13193 2.4 GHz low power transceiver for the IEEE 802.15.4 Standard, Rev. 2.8, 04/2005.
[7] Freescale Semiconductor, HSC08 microcontrollers, MC9S08GB60/D data sheet, Rev. 2.3, 12/2004.
[8] H. Karl, A. Willig, Protocols and Architectures for Wireless Sensor Networks, Wiley edition, 2005.
[9] J.N. Al-Karaki, A.E. Kamal, Routing techniques in wireless sensor networks: a survey, IEEE Wireless Communications 11 (6) (2004) 6–28.
[10] V. Rodoplu, T. Meng, Minimum energy mobile wireless networks, IEEE JSAC 8 (17) (1999) 1333–1344.
[11] R. Wattenhofer, L. Li, P. Bahl, Y.-M. Wang, Distributed topology control for power efficient operation in multihop wireless ad hoc networks, in: Proceedings of IEEE Infocom, Anchorage, AL, April 2001, pp. 1388–1397.
[12] J. Wu, F. Dai, M. Gao, I. Stojmenovic, On calculating power aware connected dominating sets for efficient routing in ad hoc wireless networks, IEEE/KICS Journal of Communications and Networks 1 (4) (2002) 59–70.
[13] C. Intagonwiwat et al., Directed diffusion for wireless sensor networking, IEEE/ACM Transaction on Networking 11 (1) (2003) 2–16.
[14] J. Heidemann, F. Silva, D. Estrin, Matching data dissemination algorithms to application requirements, in: Proceedings of ACM Sensys, Los Angeles, CA, November 2003, pp. 218–229.
[15] A. Bachir, D. Barthel, M. Heusse, A. Duda, Micro-frame preamble MAC for multihop wireless sensor networks, in: Proceedings of IEEE ICC, Istanbul, Turkey, June 2006.
[16] A. Bachir, D. Barthel, M. Heusse, A. Duda, Abstract frames for reducing overhearing in wireless sensor networks, in: Proceedings of the IFIP Networking, Coimbra, Portugal, 2006.
[17] Available from: <http://www.isi.edu/nsnam/ns/>.
[18] C.-K. Toh, H. Cobb, D.A. Scott, Performance evaluation of battery-life-aware routing schemes for wireless ad hoc networks, in: Proceedings of IEEE ICC, Helsinki, Finland, June 2001, pp. 2824–2829.
[19] A. Misra, S. Banerjee, MRPC: maximizing network lifetime for reliable routing in wireless environments, in: Proceedings of IEEE WCNC, Orlando, FL, March 2002.
[20] Qun Li, Javed A. Aslam, Daniela Rus, Online power-aware routing in wireless ad hoc networks, in: Proceedings of ACM Mobicom, Rome, Italy, July 2001, pp. 97–107.
[21] Q. Li, J. Aslam, D. Rus, Distributed energy-conserving routing protocols for sensor networks, in: Proceedings of IEEE HICSS, Big Island, HI, January 2003.
[22] R.C. Shah, J.M. Rabaey, Energy aware routing for low energy ad hoc sensor networks, in: Proceeding of the IEEE WCNC, Orlando, FL, March 2002.
[23] S.Y. Ni et al., The broadcast storm problem in mobile ad hoc network, in: Proceedings of IEEE/ACM Mobicom, Seattle, WA, August 1999.
[24] T.J. Kwon, M. Gerla, Efficient flooding with passive clustering (PC) in ad hoc networks, ACM SIGCOMM Computer Communication Review 32 (1) (2002) 44–56.
[25] Xiaoxing Guo, Broadcasting for network lifetime maximization in wireless sensor networks, in: Proceedings of IEEE SECON, Santa Clara, CA, October 2004.



Abdelmalik Bachir graduated from National Institute of Informatics, Algiers, Algeria, in 2001, received the DEA diploma in Informatics from the university of Marseille, France, in 2002, and the Ph.D. degree from Grenoble Institute of Technology, France, in 2007. From 2002 to 2004, he was a research assistant at the University of Avignon, France, working on ad hoc networking for inter-vehicle communication. From 2004 to 2007, he was a research engineer at France Telecom R&D, Meylan, France, working on medium access and routing protocols for low-power wireless sensor networks. Currently, he is a postdoctoral research fellow with at Grenoble Institute of Technology working design and evaluation of low power protocols for sensor networks.

Dominique Barthel received the degrees from Ecole Polytechnique, Paris, France, and Ecole Supérieure d'Electricité, Paris. He is currently with France Telecom, Meylan, France, where he currently leads a research project on technologies for mobile terminals and embedded devices. He is a devoted radio amateur and is the holder of six patents. His previous research interests include the architecture and design of microprocessors, digital signal processing, media processors, and scientific computers. His current research interests include networked smart devices and sensor networks, with an emphasis on low-power architectures and implementations.

Martin Heusse graduated from ENST Bretagne in 1996. He received is Ph.D. from EHESS in 2001. M. Heusse is currently assistant professor with the Joseph Fourier University in Grenoble and member of the Grenoble informatics laboratory (LIG).

Andrzej Duda is a Professor at INP Grenoble (Institut National Polytechnique) - ENSIMAG (Ecole Nationale Supérieure d'Informatique et de Mathématiques Appliquées de Grenoble). He is a member of Grenoble Informatics Laboratory. He received his Ph.D. from the Université de Paris-Sud in 1984 and his Habilitation diploma from the Grenoble University in 1994. Previously, he was an Assistant Professor at the Université de Paris-Sud, a Chargé de Recherche at CNRS, and a Visiting Scientist at the MIT Laboratory for Computer Science. In 2002-2003, he was an Invited Professor at EPFL (Swiss Federal Institute of Technology in Lausanne). He published over 80 papers in the areas of performance evaluation, distributed systems, multimedia, and computer networks. He has been on program committee of several international conferences and acts as an expert for the European Commission. His current research interests include wireless and mobile networks, spontaneous and autonomic networks, as well as sensor networks.