



HAL
open science

Adaptive Parametric Routing Based on Dynamic Metrics for Wireless Sensor Networks

Nesrine Ouferhat, Abdelhamid Mellouk, Brice Augustin

► **To cite this version:**

Nesrine Ouferhat, Abdelhamid Mellouk, Brice Augustin. Adaptive Parametric Routing Based on Dynamic Metrics for Wireless Sensor Networks. Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE, Nov 2010, Miami, United States. hal-00685682

HAL Id: hal-00685682

<https://hal.science/hal-00685682>

Submitted on 5 Apr 2012

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.

Adaptive Parametric Routing based on Dynamic Metrics for Wireless Sensor Networks

Nesrine Ouferhat, Abdelhamid Mellouk, Brice Augustin

Image, Signal and Intelligent Systems Laboratory-LiSSi Lab
Dept. of Networks and Telecoms, IUT C/V
University of Paris-Est (UPEC), Creteil, France.
Email: {[mellouk](mailto:mellouk@u-pec.fr)}@u-pec.fr

Abstract—Designing a QoS-aware, yet energy-saving routing protocol for WSNs is a notoriously hard problem. However, the outstanding interest for this technology, and the growing number of envisioned applications, motivate the need to introduce the notion of Quality of Service (QoS) in these networks. This paper introduces EDEAR (Energy and Delay Efficient Adaptive Routing), an adaptive routing algorithm based on route exploration and reinforcement learning. We evaluate EDEAR with simulations, under various network mobility conditions. Our results show that EDEAR outperforms any other routing protocol, delivering packets with the shortest delay, while reducing energy consumption. As a result, EDEAR's features allow to increase the network lifetime by 9-18%.

Index Terms—State dependent algorithm, Multi criteria routing optimization, Delay, QoS, Energy consumption.

I. INTRODUCTION

Wireless sensor networks (WSN) are a promising technology to monitor and collect specific measures in any environment. Several applications have already been envisioned, in a wide range of areas such as military, commercial, emergency, biology and health care applications.

A sensor is a physical component able to accomplish three tasks: identify a physical quantity, treat any such information, and transmit this information to a sink [1], [2], [3].

Need for QoS to guarantee the quality of real time services. Take into account not only the static network parameters but also the dynamic ones.

Therefore, QoS measures should be introduced to the network so that quality of real time services can be guaranteed. The most popular formulation of the optimal distributed routing problem in a data network is based on a multicommodity flow optimization whereby a separate objective function is minimized with respect to the types of flow subjected to multicommodity flow constraints. Given the complexity of this problem, due to the diversity of the QoS constraints, we focus our attention in this

paper on bio-inspired QoS routing policies based on the Reinforcement Learning paradigm.

Many research works focus on the optimization of the energy consumption in sensor networks, as it directly affects the network lifetime [1], [2], [4]. Routing protocols were proposed to minimize energy consumption while providing the necessary coverage and connectivity for the nodes to send data to the sink [2], [3]. Other routing protocols have also been proposed in WSN to improve other QoS constraints such as delay.

The complexity of routing protocols increases dramatically with the integration of more than one QoS parameter. Indeed, determining a QoS route that satisfies two or more constraints (for example, delay and cost) is an NP-complete problem [5], because the Multi-Constrained Optimal path problem cannot be solved in polynomial time. Therefore, research focus has shifted to the development of pseudo-polynomial time algorithms, heuristics, and approximation algorithms for multi-constrained QoS paths.

This paper proposes a new adaptive routing protocol that simultaneously improves two dynamic criteria: energy consumption and end-to-end delay. Our algorithm, called EDEAR (Energy and Delay Efficient Adaptive Routing) is based on a continuous time evolving mechanism. It uses the sink to proactively explore the network in search of alternate, better paths to transmit data packets. Each node maintains multiple paths to the sink and selects, in a probabilistic way, the best path according to a multicriteria cost function which allows to minimize both the network delay and the energy consumption.

We evaluate EDEAR against three state-of-the-art routing protocols. We stress them by varying mobility network parameters. For each scenario we report the mean end-to-end delay, energy consumption and delivery rate. Our simulation with NS-2 show that EDEAR outperforms all other protocols for each stress condition, demonstrating its ability to quickly and efficiently select routes offering both a good delay and a minimum energy consumption. As a result, EDEAR increases the network

by 9-18% compared to EAR.

Our paper is organized as follows: Our proposal is detailed in next section. Simulations are given in section 3 and we will finish by conclusion in last section.

II. EDEAR PROPOSAL

As wireless sensors networks are characterized by dynamic traffic conditions, this requires taking into account the level of QoS routing process. Therefore, each approach of adaptive routing must be robust and reactive enough to accommodate changes in traffic conditions while minimizing the packet delivery time. Approaches of routing based on reinforcement learning have been proposed recently. This type of approach is well suited to the problem of QoS routing, as the environment model where node is located is a priori unknown. However, the effectiveness of these approaches in routing decision depends highly on conditions of current traffic on the network and its evolution.

Our proposal is based on adaptive approaches; its objective is to find the best path in terms of energy consumed and end-to-end delay. It assumes that at each node, information about the residual energy, the energy consumed and the average delay links are available even before the route is requested. We assume that these estimates are made during the discovery of neighbor by an external mechanism independent of routing process.

Our proposal is also considered as hybrid protocol; it combines the concept of on demand searching routes, and proactive exploration concept. This joint mechanism in the exploration phase allows to EDEAR to find alternative routes missed in the first phase. For this, EDEAR is based on two agent explorers: *Int-E-Route* (Interest Exploration Route) and *Resp-E-Route* (Response Exploration Route). The first one, generated at the sink when this one demands a request, is available on the network to find routes to the source node. To limit overhead generated by packets discovery, we use an optimized distribution mechanism based on Multi Point Relay (MPR) OLSR protocol [6].

Neighbours list obtained via Hello messages (extension of OLSR)

Optimization of the transmission of exploration packets (MPRE) (extension of MPR).

Two explorer agents, one sent by sink to node, the other sent by node in response to the first one.

The arrival agent *Int-E-Route* to the source node initiates the creation of the second agent, *Resp-E-Route*. This latter takes the reverse path followed by the first *Int-E-Route* agent.

A. *Int-E-Route* Agent

We use the mechanism of periodic “Hello” messages to discover the neighbors. Each node broadcasts Hello messages, containing a list of its neighbors and the links state with its symmetrical neighbors, asymmetrical or lost. “Hello” messages are received by its one hop neighbors and will not be relayed; each node is able to know its one

and two hop neighbors. A message is sent in broadcast way and will not be relayed (the distribution is local). This latter is composed by:

- A list of addresses of its symmetrical neighbors.
- A list of addresses of its asymmetrical neighbors.

When a “Hello” message is received by a neighbor, if a node finds its own address in the asymmetrical list, then, this link is valid.

A node is considered as a neighbor if and only if the collision rate of “Hello” packets sent to it, is below a certain threshold fixed initially.

Whenever, *Int-E-Route* gets through intermediate node, it retrieves its address and the information of its residual energy, energy consumed on link and the average delay.

There is a neighbour table in each node of the network. This table, shown in fig. 1, contains:

- N-id: ID of its one-hop neighbours.
- N-states: link state with its one hop neighbors (symmetric or asymmetric).
- List-N-2 hops: its two-hop neighbors.
- E_r : Residual energy of node.
- E_c : Energy consumed on each link by one-hop neighbors.
- D_{link} : Link delay.
- QE_c : Cost energy consumed on the link.
- QD_{link} : Cost delay of the link.

N-ID	N-states	N2-ID	E_r	E_c	D_{link}	QE_c	QD_{link}
------	----------	-------	-------	-------	------------	--------	-------------

Figure 1. Neighbors Table.

Packets *Int-E-Route* are sending periodically as the route to this node is requested. This period is defined by the interval exploration δ which is calculated by simulation (for our scenarios, this value started with 10 seconds). Thus, each δ times, a new packet *Int-E-Route*, with new sequence number, is created and sent in the same manner as first *Int-E-route* is generated.

This agent is sent in broadcast to all one-hop neighbors nodes. To reduce overhead generated by packets exploration, only MPRE nodes will be allowed to relay this packet.

Int-E-Route recovers energy consumed and delay with its neighbors and stores it in list memory. The sequence number of packet does not change. This process continues until *Int-E-Route* arrives to the source node.

The cost path from the source to sink is calculated as follow:

Let's the residual energy of each node must be exceeds a parameter ϵ fixed before (ϵ corresponds to the energy required to send a fixed volume of information):

$$E_r \geq \epsilon \quad (1)$$

Where:

E_r : represent the residual energy and \mathcal{E} the threshold of minimum energy for active node before sending all data.

The cost of a node i for a link between nodes i and j is computed as:

$$\text{Cost of node } i^j = \text{Cost of link } i^j = \gamma QE_{c_i}^j + \theta QD_i^j \quad (2)$$

Where:

QE_c is the cost of energy consumed on link (i, j) , and QD represent the cost of mean delay on link (i, j) , γ and θ are tuneable parameters, with $\gamma > \theta$ to give more importance to the energy metric.

The cost function can be completed as the cost of path as follows:

$$f_{\text{cost}} = \sum_{j=1}^K h_j C_j \quad (3)$$

C is a scalable vector for QoS metrics, and K represent the number of QoS criteria considered in routing process.

The cost of the whole path constructed with N nodes between source node and sink is:

$$\text{Cost of path} = \gamma \sum_{i=1}^{N-1} QE_{c_i} + \theta \sum_{i=1}^{N-1} QD_i \quad (4)$$

B. EDEAR's exploration mechanism

In our EDEAR proposal, we used the same mechanism as OLSR [6] by replacing the cost function by the energy cost stored in the neighbors table (fig. 1). So, the choice of MPRE for each node will be based on the links that offer the best cost estimated locally. The algorithm for selecting MPRE should be summarized as follows:

- Each node N_i of the network as MPRE chooses among its neighbors for one-hop, all the nodes up to a neighbor with two-hops.
- Each node selects as MPRE node that has the best link cost. In the case of equality between two nodes, the one that covers more than two-hop neighbors is chosen. This step is repeated until all neighbors at two hops are covered.

C. EDEAR's Resp-E-Route Agent

Agent *Resp-E-Route* updates routing tables based on information in the neighbors tables over all nodes crossed. This update consists of readjustment of link costs associated with each entry in routing table and generated by source node upon reception of *Int-E-Route* packet. It retrieves all information gathered by the packet *Int-E-Route* at the sink. The Packet *Resp-E-Route* is sent in unicast mode and takes the opposite route borrowed by the packet *Int-E-Route*. The source node generates packets *Resp-E-Route* on the basis of information on total cost of path followed by the packets *Int-E-Route*. When the arrival of the first packet *Int-E-Route* to source node is happen, the variable "best cost path" is used to store the

cost of the path. At each arrival *Int-E-Route* packet to the source node, the cost of path calculated for this route is compared with the lowest cost path saved in the variable "best cost route". If there is a better path, the traffic will switch to this one. Each *Resp-E-Route* packet retraces exactly the same path taken by the *Int-E-route* packet which is at the origin of the path. At each passage trough the intermediate node, it compares its address with the destination address of sink *Resp-E-Route* packet to verify if it reaches the destination.

D. EDEAR's updating and calculating routing table

Routing process consist to find the path with minimum cost in order to minimize simultaneously node energy consumed on all paths and the end-to-end delay between sink and source node. The fundamental problem is to update the link cost; the algorithm that we propose is adaptive and is based on routing tables maintained at each node. When a route is requested, the node controls its routing table if there is an entry that corresponds to requested node. If no entry appears for that node, the node creates an entry and initializes the cost of links; evenly distributed over all its neighbors MPRE.

These costs are updated during transition from the agent *Resp-E-Route* based on information of links cost retrieved by visiting all nodes on the route. Then, we replace in each link:

$$\text{Cost of node } i^j = \text{Cost of link } i^j = \gamma QE_c + \theta QD \quad (5)$$

The new costs at $(t + \delta t)$ in the node i on the link (i, j) is:

$$QE_{c_i}^j(t+1) = \frac{E_{c_i}^j(t)}{E_{c_i}^j(t+\delta)} * QE_{c_i}^j(t) \quad (6)$$

and

$$QD_{c_i}^j(t+1) = \frac{D_{c_i}^j(t)}{D_{c_i}^j(t+\delta)} * QD_{c_i}^j(t) \quad (7)$$

Where:

$E_{c_i}^j(t)$: Energy consumption in the node i on the link (i, j) at a time t .

$E_{c_i}^j(t + \delta t)$: Energy consumption in the node i on the link (i, j) at a time $(t + \delta)$.

$D_{c_i}^j(t)$: Link delay in the node i on the link (i, j) at a time t .

$D_{c_i}^j(t + \delta t)$: Link delay in the node i on the link (i, j) at a time $(t + \delta)$.

We refresh then the new cost of the path based on equation (4) and continues iteratively the update process.

The variable "Best cost route" is updated as follow:

$$\text{'Best cost route'} = \text{Max} \left(\gamma \sum_{i=1}^{N-1} QE_{c_i} + \theta \sum_{i=1}^{N-1} QD_i \right) \quad (8)$$

III. SIMULATION AND RESULTS

This section first describes our simulation setup and the metrics we used to evaluate EDEAR. Then, it presents the results obtained under three scenarios designed to compare EDEAR with three other protocols: QoS_AODV [7], SAR [8] and EAR [9].

QoS_AODV: The link cost used in this protocol is formulated as a function which takes into account node's energy consumed and error rate. Based on extended version of Dijkstra's algorithm, the protocol establishes a list of paths with minimal cost. Then, the one with the best cost is selected. This protocol uses hierarchical routing; it divides the network into clusters. Each cluster consists of sensor nodes and one cluster head. The cluster head retrieves data from the cluster and sends it to the sink. QoS routing is done locally in each cluster. We modified QoS_AODV in order to use the energy metric for the choice of the paths. Our extension takes into account the end-to-end delay and replaces the bandwidth metric with the energy consumed along the path.

SAR (Sequential Assignment Routing): SAR manage multi-paths in a table routing which attempts to achieve energy efficiency and fault tolerance. SAR considers trees QoS criteria during the exploration of routing paths, the energy resource on each path and priority level of each packet. By using these trees, multiple paths of the sensors are well trained. One of these paths is chosen depending on energy resources and QoS of path. SAR maintains multiple paths from nodes to sink. High overhead is generated to maintain tables and states at each sensor.

EAR (Energy Aware Routing): EAR is a reactive routing protocol designed to increase the lifetime of sensor networks. Like EDEAR, the sink searches and maintains multiple paths to the destinations, and assigns a probability to each of these paths. The probability of a node is set to be inversely proportional to its cost in terms of energy. When a node routes a packet, it chooses one of the available paths according to their probabilities. Therefore, packets can be routed along different paths and the nodes' energy will be fairly consumed among the different nodes. This technique keeps a node from being over-utilized, which would quickly lead to energy starvation.

A. Simulation setup and metrics

We use NS-2 for our simulations. We randomly distribute 100 nodes on a 200x200m area. The nodes send 35 bytes messages at a maximum of 200 kbps. The initial energy for each node is 5J. For each scenario, we run 10 simulations with a topology chosen at random. Each run lasts 300 seconds. Finally, we set γ to 0.75 and θ to 0.25, to give more importance to the energy metric.

For the evaluation, we stress the four protocols by varying mobility network conditions. Varying each network parameter allows to study a different feature of the routing algorithms. Highly mobile nodes will test the algorithm's ability to quickly detect route failures and establish new ones. For each mobility scenario, we

increase one of these network parameters to force network congestion and energy starvation. We expect that the different protocols will react and behave differently to protect the network's lifetime, while guaranteeing a good QoS.

To this end, we are interested in three metrics to evaluate EDEAR and compare it to the other protocols. These metrics were carefully chosen in order to give a broad idea of the improvements achieved by our protocol:

- The *Energy consumption* represents the energy (in mJ) spent by the nodes to route the packets to the sink. It includes both data packets and signalling packet. Energy consumption has a direct impact on the lifetime of a sensor network.
- The *End-to-end delay* is the time for a packet to reach the sink. It includes the time spent to discover the routes, as well as the time spent in the nodes queues and in the air. This metric represents the global response time of the network, and depends on the protocol efficiency in terms of the number of congested nodes and stability.
- The *Delivery rate* is the number of lost packets divided by the number of packets sent. This metric represents the reliability of the protocol in terms of packet delivery.

B. Results

Our study concerns the evaluation of the protocols for an increasing mobility of the deployed sensors. The first case shows the protocol's behaviour when nodes are completely static (0 m/s), and we consider extreme cases where nodes move at up to 30 m/s.

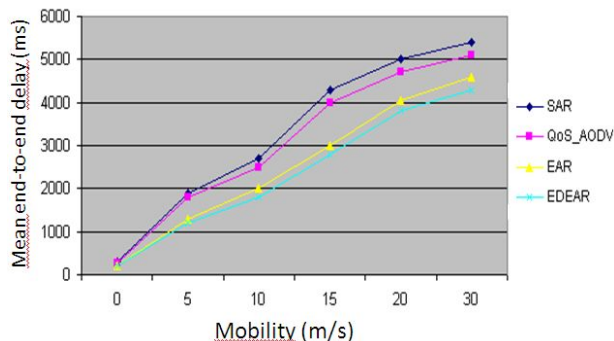


Figure 2. Mean delay for various mobility patterns

Fig. 2 plots the mean end-to-end delay achieved by all four protocols under various speeds of the nodes. In the case of a static topology, the results for all protocols are quite similar. However, increasing the mobility reveals a clear difference in the protocols' efficiency. The plot indicates that, for speeds above 5 m/s, the performances of QoS_AODV and SAR degrade quickly (the delay increases from 2000 to 5000 ms), while EAR and EDEAR keep delivering packets on a reasonable delay, for a speed up to 10 m/s. Still, our protocol resists better to increasingly harder conditions.

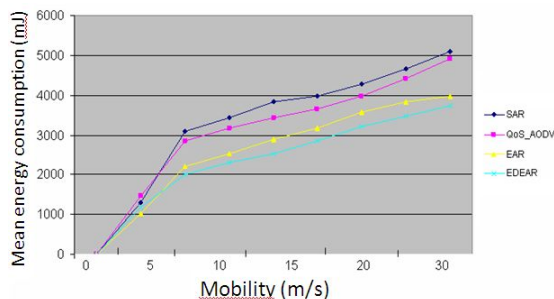


Figure 3. Mean energy consumption for various mobility patterns

When nodes are mobile, topology changes are frequent. These changes cause a frequent loss of the currently selected best paths, therefore triggering route discovery phases more frequently. Discovery phases generate additional control traffic, consuming more energy and bandwidth. As a result, resources for data traffic are reduced, and the nodes quickly consume their energy. To study this effect, Fig. 3 plots the energy consumption when nodes move at various speeds. The dynamics clearly have less impact on EDEAR, which takes advantage of its improved reactive capabilities when compared to the other protocols. Indeed, the choice of both energy-saving and low delay nodes dramatically reduces the energy consumption. Routing tables are probabilistic in EDEAR; several paths may be selected to reach a destination. Therefore, if a node's neighbours change, the mechanism of adjusting probabilities can be adapted almost instantly to degradation performance on the selected path. Furthermore, nodes mobility can lead to the creation of new routes involving nodes with a high residual energy. EDEAR quickly detects and selects these high quality nodes, thus allowing it to resist more efficiently to frequent route changes.

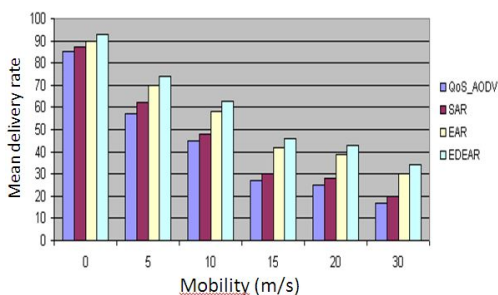


Figure 4. Mean delivery rate for various mobility patterns

Finally, Fig.4 considers the mean delivery rate achieved when the mobility increases. Under a static scenario, all protocols achieve a very good delivery rate, EDEAR being the only one higher than 90%. As expected, the delivery rate dramatically decreases with an increasing mobility, but the decrease is less significant for adaptive protocols, especially for EDEAR. For a speed of 10 m/s, EDEAR still achieves a delivery rate greater than 60%, while SAR and QoS_AODV are well below 50%. Packet losses occur more frequently as the network dynamicity increases, because routing protocols have to face the problem of recomputing new routes all the time. During this phase, no routes are known to reach the

destination, and the nodes have no choice but dropping the packets. However, thanks to EDEAR's exploration process and its probabilistic nature, it is able to quickly reallocate a new route, thus decreasing the time when no route is available.

IV. CONCLUSION

In this paper, we presented a new way to make adaptive routing with quality of service in order to improve the end-to-end delay and increase the lifetime of a delay tolerant network. Our protocol, called EDEAR, explores the network and chooses the best path routing in terms of energy and delay to route information based on reinforcement learning paradigm. For that, an explorer agent is used and learns from past experiences to choose the next path; data is always sent over the best path.

Our approach offers advantages compared with other classical approaches, such as SAR protocol. In particular, it reduces much better the energy consumed required for information sent, updates routing and network exploration more reactively. To reduce the overhead packets in the exploration process, our protocol used an enhanced version of OLSR algorithm based on energy to construct MPRE nodes.

Simulation results have shown the efficiency of the adaptive approaches compared to traditional approaches. In fact, the proposed algorithm takes into account the network state in a better way than the classical approaches do. Also, our protocol is well-suited for a dynamic mobile environment. In particular, an adaptive algorithm that simultaneously processes an observation and learns to perform better is a good way to consider time evolving networks. Such adaptive algorithms are very useful in tracking a phenomenon that evolves over time, especially in Delay Tolerant Networks.

Finally, our work in progress consists to use other metrics to find the best optimal paths (residual bandwidth, loss ratio, etc.) and to split traffic under all the considered paths using cognitive approaches.

REFERENCES

- [1] A. Kumar, D. Manjunath, J. Kuri, D. Manjunath, "Ad Hoc Wireless Sensor Networks (WSNs)". *Wireless Networking*, 2008, pp. 337-37.
- [2] J-F. Buford, H.Yu, E-K.Lua, "Mobility and Heterogeneity", *P2P Networking and Applications*, 2009, pp. 299-317.
- [3] R. Olfati-Saber, A. Fax, and R.M. Murray, "Consensus and cooperation in networked multi-agent systems", *Proceedings of the IEEE Proceedings*, vol. 95, no. 1, 2007, pp. 215-233.
- [4] S. Farahani, "Battery Life Analysis ZigBee", *Wireless Networks and Transceivers*, 2008, pp. 207-224.
- [5] A. Mellouk, S. Hoceini, M. Cheurfa, "Reinforcing probabilistic selective Quality of Service routes in dynamic irregular networks", *Computer Communications*, vol. 31, no. 11, 2008, pp. 2706-2715.
- [6] D-Q. Nguyen, P. Minet, "Analysis of MPR selection in the OLSR protocol". *Proc. Of PAEWN*, Niagara Falls, Ontario, Canada, 2007.
- [7] C, E, Perkins. E, M, Royer. S, R, Das. "Quality of Service for Ad hoc On-Demand Distance Vector Routing". *IETF Internet Draft*, work in progress, July 2000.
- [8] C. Ok, S. Lee, P. Mitra, S. Kumara, "Distributed energy balanced routing for wireless sensor networks", *Computers & Industrial Engineering*, vol. 57, no. 1, 2009, pp. 125-135.
- [9] R. C. Shah, J. M. Rabaey. "Energy aware routing for low energy ad hoc sensor networks". *Proceeding of IEEE Wireless Communications and Networking Conference (WCNC)*, Orlando, FL, March 2002.