Performance Sensitivity of Routing Algorithms with Various Models of Wireless Sensor Networks

Julien Bernard #1, Violeta Felea #2

Institut FEMTO-ST UMR CNRS 6174 – University of Franche-Comté
16 route de Gray, 25030 Besançon CEDEX, France
1 julien.bernard@femto-st.fr
2 vafelea@femto-st.fr

***Abstract*—Routing algorithms for wireless sensor networks are often evaluated through simulations in order to measure the lifetime of the network and the efficiency of the algorithm regarding energy consumption. Some particular communication models with their parameters' set are used to implement the simulation and are rarely identical. We exhibit two kinds of performance sensitivity regarding simulations: the first one concerns the communication model itself and the second one is generated by the parameters of the communication model. We provide a generalized model that covers very different situations and we state the routing problem as a linear programming problem in order to measure the absolute efficiency of the algorithms with different models and parameter values. Our experiments run using two routing protocols, LEACH and Flow Augmentation, showed that different models or parameters can lead to significantly different results and conclusions. We tried to characterize the origin of the performance sensitivity in each case.**

I. Introduction

A wireless sensor network is a network composed of sensor nodes communicating wirelessly. A node can get a measure from its sensor (temperature, pressure, etc) and send it through its wireless communication device. The node communicates with all the nodes that are in the range of its wireless communication device, forming a graph. A source node is a node that generates information while a sink node is a node that gathers information.

The main challenge in wireless sensor networks is energy saving as the nodes generally work with a limited battery. The major part of energy consumption comes from communications, when the nodes send and also receive messages. So the challenge is to optimize the communication scheme in order to save energy and hence maximize the lifetime of the network.

Routing algorithms for wireless sensor networks have been designed to compute paths from sources to sinks while minimizing energy consumption. They are often evaluated through simulations in order to measure the lifetime of the network and the efficiency of the algorithm regarding energy consumption. The models and the parameters for communications are often fixed once and for all and are rarely discussed.

We show that different models and parameters used within the same algorithm can lead to different results and conclusions. We exhibit two sources for the performance sensitivity of routing algorithms. The first one concerns the communication model itself, especially the messages that are taken into account in the energy consumption. The second one is generated by the parameters of the communication model.

In section II, we analyze different communication models and parameters of the literature. Then, in section III, we provide a generalized model for communications that can be reduced to cover a wide range of situations. We state the routing problem as a linear programming problem in order to measure the absolute efficiency of the algorithms with different models and parameters. In section IV, we run some experiments on two different routing algorithms, LEACH [1], a cluster based protocol, and Flow Augmentation [2], a shortest-path based algorithm, in different models to show the performance sensitivity of the algorithms.

II. Models and Parameters from Previous Works

In this section, we analyze previous works and discuss a set of models and a set of parameters found in the literature.

A. Communication Models

Due to wireless communications, when a node sends a message all its neighbors may receive it, whereas only one of them is the real recipient. We call unintended messages the messages that are received at a node when the node is not the real recipient. Overhearing can alter the relative efficiency of the algorithm [3].

Many previous works state the routing problem in wireless sensor networks as a linear programming problem. Then, they generally propose an algorithm to find a solution as close as possible to the optimal solution. These works differ in the model they use.

We define three energy models for the communications, that cover a wide range of situations regarding the MAC layer and the requirements of applications:

- *Sender-Only model*: this model only considers the energy for sending messages. It is the simplest possible model. The energy for receiving is set to 0, for either intended or unintended messages.
- *Sender-Receiver model*: this model considers the energy for sending messages and for receiving intended messages but not the energy for receiving unintended messages.
- *Neighborhood model*: this model is the most complete as it considers the energy for sending and receiving both

types of messages, intended or unintended. We will use this model that can easily be reduced to the previous ones.

Sankar and Liu [4] use the Sender-Only model. They give a local-control flow algorithm with a performance guarantee that works for static and slowly varying networks. Madan, Luo and Lall [5] give a distributed algorithm that approximates the linear programming problem and converges linearly. They also use a Sender-Only model and assume the scheduling of the messages involves no interference. Madan and Lall [6] later give two distributed subgradient algorithms to solve the same problem.

Chang and Tassiulas [2] consider the Sender-Receiver model. They propose an algorithm called Flow Augmentation (FA), based on table-driven routing, that has good performance in this model. We have implemented this algorithm in our experiments to make a comparison with their results but using different models and parameters. Xue, Cui and Nahrstedt [7] propose a non-linear utility-based optimization formulation, and derive a fully distributed routing algorithm. They use the same Sender-Receiver model as [2].

Park and Sahni [8] use a Sender-Only model. They prove that the problem is NP-Hard and develop an online heuristic for routing.

B. Parameters for Energy Consumption

The energy consumption for the sending of a message by node i to node j involves two parts: the energy spent by node i for transmitting, noted e_{ij}^T, and the energy spent by node j for receiving, noted e_{ij}^R. Heinzelman et al. [1] provided a first order radio model that has been widely used since then:

$$\begin{cases} e_{ij}^T = \lambda \times (e^T + \epsilon_{\text{amp}} d_{ij}^\alpha) \\ e_{ij}^R = \lambda \times e^R \end{cases}$$

where λ is the number of bits in the message, e^T and e^R is the energy dissipated in the circuitry, ϵ_{amp} is the transmit amplifier ratio, d_{ij} is the distance between node i and node j and α depends on the channel model ($\alpha = 2$ for free space and $\alpha = 4$ for multipath). In their original publication, they give a set of parameters for this model: $e^T = 50$ nJ/bit, $e^R = 50$ nJ/bit, $\epsilon_{\text{amp}} = 100$ pJ/bit/m^2 and $\alpha = 2$.

Later, this model has been used in [2] with a different set of parameters which differs from [1] in two ways.

First, the energy for receiving e^R is set to 150nJ/bit whereas it is set to 50nJ/bit in [1]. This implies that the energy for receiving is always higher than the energy for sending if $d <$ 5m. This difference is justified in [2] by the complexity of the receiving circuitry. Santos et al [9] show that this assumption is not always true: for example, MicaZ consumes more when receiving (factor: 1.5) whereas Mica2 consumes more when sending (factor: 2).

Second, the exponent α in the energy for sending has been set to 4 instead of 2. These values are classical in the literature: 2 is used for free spaces and 4 is used in case of multipath reflections. But in this case, the value of ϵ_{amp} should also have changed, as it is expressed in pJ/bit/m^2 in [1]. Heinzelman et al. give a mixed model in [10], using a free space model ($\alpha = 2$) for distances smaller than d_0, with $\epsilon_{\text{fs}} = 10$pJ/bit/m^2 and a multipath model ($\alpha = 4$) for distances greater than d_0, with $\epsilon_{\text{mp}} = 0.0013$pJ/bit/m^4.

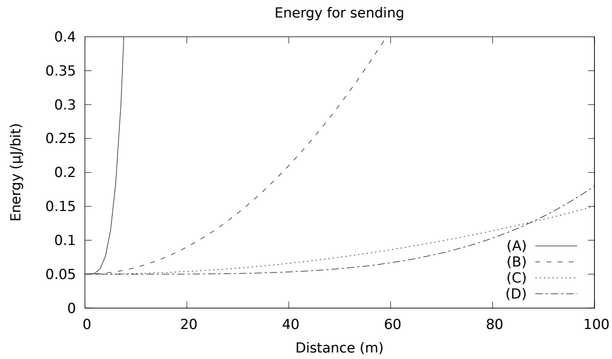

Fig. 1. Comparison of energy for sending e_{ij}^T with different parameters detailed in table I

TABLE I
PARAMETERS USED FOR THE FIRST ORDER RADIO MODEL IN PREVIOUS WORKS

	α	ϵ_{amp}	e^T	e^R	Ref.
(A)	4	100pJ/bit/m^4	50nJ/bit	150nJ/bit	[2]
(B)	2	100pJ/bit/m^2	50nJ/bit	50nJ/bit	[1]
(C)	2	10pJ/bit/m^2	50nJ/bit	50nJ/bit	[10]
(D)	4	0.0013pJ/bit/m^4	50nJ/bit	50nJ/bit	[10]

Figure 1 shows the energy for sending with these different parameters that are detailed in table I, relative to the distance.

III. GENERAL MODEL AND PROBLEM DEFINITION

In this section, we provide a generalized model for communications that can be reduced to cover a wide range of situations. As we change the models and parameter values in the experiments, we need an absolute measure of the performance of the algorithms. We can not compare the results in different models or with different parameter values. So, we choose to measure the performance of the routing algorithms against the optimal solution given by the linear programming problem that is described in this section.

A. Model of Wireless Sensor Networks

A wireless sensor network is modeled by a directed graph $G = (V, E)$, with V the set of vertices representing the sensor nodes, and E the set of edges representing the link between nodes, i.e. if i and j are two nodes in V, $(i, j) \in E$ means that i can send a message to j, or j is in the radio range of i. The reciprocal is not necessarily true. d_{ij} is the euclidian distance between i and j.

The set of nodes N_i that can be reached by i is called the *neighborhood* of i. $N_i = \{j \in V, (i, j) \in E\}$. We also define $N_i^{(j)+}$, respectively $N_i^{(j)-}$, the set of nodes in the neighborhood of i that are farther, respectively closer, than node j. So we have: $N_i^{(j)+} = \{k \in N_i, d_{ik} \geq d_{ij}\}$ and $N_i^{(j)-} = \{k \in N_i, d_{ik} \leq d_{ij}\}$.

For the communication model, we use the Neighborhood model which is the most general in the sensor network context. Moreover, in order to compute the optimal solution, we consider that a node always uses the exact necessary amount of energy to reach its expected destination. When a message is sent from i to j, with j the intended destination of the message, i consumes the exact energy for sending the message to distance d_{ij}, and all the nodes in $N_i^{(j)-}$ consume some energy for receiving this message. This is a generalization of previous models that can be reduced to cover weaker models.

We suppose that node i has an initial amount of energy E_i. It consumes an energy e_{ij}^T for transmitting a message to node j and e_{ki}^R for receiving a message from node k, either intended or unintended.

B. Maximum Lifetime Routing Problem

Now that we have set the model, we define the problem by extending previous definitions ([2], [6]) in the Neighborhood model.

We note O, respectively I, the set of source nodes, respectively sink nodes, in the network.

We define the variables x_{ij} as the number of messages sent by i where j is the intended destination of the message. Node j will transmit the message to another node unless it is a sink node, i.e. $j \in I$. A variable x_{ij} is defined for each edge in the graph and is an integer variable.

We also define the variable M as the number of messages sent by any source node. This number does not depend on the source node as we target a uniform knowledge of the area. We define $M_j, j \in I$, the number of messages received by the sink node j.

The goal is to maximize M with the following constraints.

The first constraint is the *conservation of messages*: all the messages received by every node should be transmitted, i.e. the set of incoming messages should be equal to the set of outgoing messages. For every node i that is not a sink or a source:

$$\forall i \notin O \cup I, \sum_{k, i \in N_k} x_{ki} = \sum_{k \in N_i} x_{ik} \tag{1}$$

In addition, for every node i that is a source, the M messages it generates are transmitted:

$$\forall i \in O, \sum_{k, i \in N_k} x_{ki} + M = \sum_{k \in N_i} x_{ik} \tag{2}$$

In addition, for every node i that is a sink, M_i messages are gathered.

$$\forall i \in I, \sum_{k, i \in N_k} x_{ki} = \sum_{k \in N_i} x_{ik} + M_i \tag{3}$$

Then, the second constraint is the *consumed energy*: a node i can not consume more energy than its initial energy E_i. Node i consumes energy E_i^T when it transmits messages:

$$E_i^T = \sum_{k \in N_i} x_{ik} e_{ik}^T \tag{4}$$

Node i consumes energy E_i^R when it receives a message that it must forward, and when it receives unintended messages because it is in the range of the emitter, but it is not the destination:

$$E_i^R = \underbrace{\sum_{k, i \in N_k} x_{ki} e_{ki}^R}_{i \text{ is the destination}} + \underbrace{\sum_{k, i \in N_k} \sum_{l \in N_k^{(i)+} \setminus \{i\}} x_{kl} e_{ki}^R}_{i \text{ is not the destination}} \tag{5}$$

We can combine both terms of equation (5) and we obtain:

$$E_i^R = \sum_{k, i \in N_k} \sum_{l \in N_k^{(i)+}} x_{kl} e_{ki}^R \tag{6}$$

And the constraint on energy is $E_i^T + E_i^R \leq E_i$:

$$\forall i \in V, \sum_{k \in N_i} x_{ik} e_{ik}^T + \sum_{k, i \in N_k} \sum_{l \in N_k^{(i)+}} x_{kl} e_{ki}^R \leq E_i \tag{7}$$

Then, we must set an additional constraint on the *message delivery*. Every message sent by a source should reach a sink. If s is the number of sources:

$$s \times M = \sum_{j \in I} M_j \tag{8}$$

Finally, we set the objective to the maximization of M.

The problem of maximum lifetime routing using the Neighborhood model for energy consumption is NP-Hard. Trivially, it is a generalization of the problem in the Sender-Only model which has been proved NP-Hard by Park et al. [8].

IV. Experiments and Analysis

In this section, we present two experimental scenarios from previous works. First, in section IV-A, we reproduce the experiments in [2], and especially the Flow Augmentation algorithm. This algorithm is a heuristic for routing messages that shows good performance in the Sender-Receiver model. It is interesting to compare the results of these experiments in the Neighborhood model and with different parameters. Second, in section IV-B, we consider the well-known LEACH protocol [1], [10] and compute the efficiency of the protocol, compared to the optimal solution using the Neighborhood model.

For these experiments, we have developed a simulator based on the Boost Graph Library (BGL) for the network models and graph algorithms, and the GNU Linear Programming Kit (GLPK) for the resolution of integer linear programming problems. For every experiment, we ran our simulator on 200 randomly chosen networks. When we obtained a solution to the integer linear programming problem, we ran each algorithm and computed the performance relatively to the optimal solution. Generally, more than 160 networks fulfilled the condition. Then average values and worst case values were computed from all the values and are represented in the following figures.

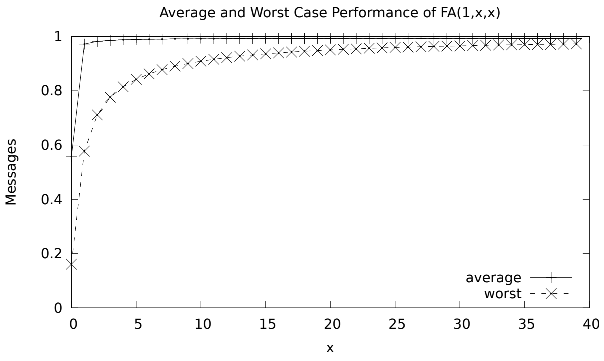

Fig. 2. Average and worse performance of FA(1,x,x) relative to the optimal solution in the Sender-Receiver model with $e^R = 150$nJ/bit, $\alpha = 4$ and $\epsilon_{\text{amp}} = 100$pJ/bit/m^4

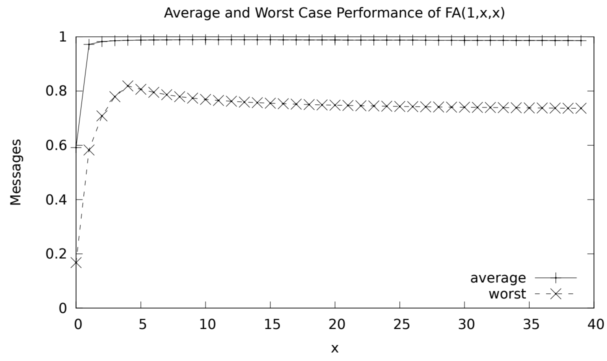

Fig. 3. Average and worse performance of FA(1,x,x) relative to the optimal solution in the Neighborhood model with $e^R = 150$nJ/bit, $\alpha = 4$ and $\epsilon_{\text{amp}} = 100$pJ/bit/m^4

A. *Flow Augmentation Algorithm*

1) Description of the Algorithm: In this section, we remind concisely the Flow Augmentation algorithm and the experiment as it can be found in [2].

The Flow Augmentation (FA) algorithm computes the shortest paths from every source to a sink and then routes an amount of λ bits from sources to sinks, until a node runs out of energy. The cost C_{ij} of link (i, j) to compute the shortest paths is given by (Eq. (10) in [2]):

$$C_{ij} = (e_{ij}^T)^{x_1} \frac{E_i^{x_3}}{\underline{E}_i^{x_2}} + (e_{ij}^R)^{x_1} \frac{E_j^{x_3}}{\underline{E}_j^{x_2}}$$

where x_1, x_2, x_3 are parameters of the FA algorithm, and $\underline{E}_i$, respectively $\underline{E}_j$, is the residual energy on node i, respectively j. e_{ij}^T and e_{ij}^R are computed with the set of parameters (A) in table I.

The results presented in [2] with different sets of (x_1, x_2, x_3) show that the best configuration is $x_1 = 1$ and $x_2 = x_3 = x$. So we only consider this set of parameters and compute FA$(1, x, x)$ for $0 \leq x < 40$.

The experiment is done on a network composed of 20 nodes, randomly distributed in a square of 50m $\times$ 50m. The connectivity model is a Unit Disk Graph [11], i.e. the neighborhood of i is defined by all the nodes that are in the disk of radius $d = 25$m around i. This model is widespread for evaluating applications in wireless sensor networks. The initial energy E_i of node i is 10J if i is even and 20J if i is odd. In the experiment, there is only one source, chosen randomly among the nodes and one sink located at $(45, 45)$.

We consider messages with a size of λ bits. $\lambda = 5000$ in the reference experiment which is quite big for a single message but necessary for the comparison.

2) First Results: Figure 2 shows the normalized number of messages, in average and worst case, in the Sender-Receiver model with the original set of parameters. This figure can be compared to figure 12 in [2] as it is the same experiment with the same parameters. We observe a great similarity in the results which validates our simulator and our implementation.

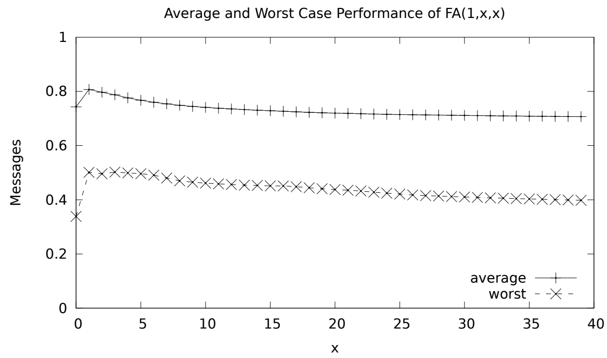

Fig. 4. Average and worse performance of FA(1,x,x) relative to the optimal solution in the Neighborhood model with $e^R = 50$nJ/bit, $\alpha = 2$ and $\epsilon_{\text{amp}} = 100$pJ/bit/m^2

If we change the model to the Neighborhood model, we have quite similar results. Figure 3 shows that even if the average performance is roughly the same, the worst case goes down to less than 80% of the optimal solution, for $x \geq 6$. In the Sender-Receiver, the worst case was more than 90% of the optimal solution, for $x \geq 9$.

As a conclusion for these first results, we can say that the FA algorithm has a worse performance in the Neighborhood model than in the Sender-Receiver model. But the difference does not seem to be significant and the algorithm is very close to the optimal solution in average. Only the worst case differs.

In this case the algorithm, in average, is not sensitive to the model. This can be explained by the nature of the algorithm. The computation of the link cost $C(i, j)$ takes into account the energy of both the sending and receiving nodes. So, even if the FA algorithm does not take unintended messages into account, it adapts to the energy that has already been spent by previous messages every time the shortest paths are computed.

3) Change of Parameter Values: Then, we keep the Neighborhood model and we change the parameters from the set of parameters (A) to the set of parameters (B) in table I. This implies two differences: $\alpha = 2$ instead of 4 and $e^R = 50$nJ/bit instead of 150.

Figure 4 shows a significant difference from the previous results. The number of messages delivered by the FA algorithm is only 80% of the number of messages of the optimal solution on average. The worst case is around 50% of the optimal solution. This shows that the FA algorithm is very sensitive to the energy parameters.

The reason for such a change in the results is in the parameters. With the set of parameters (A), we can see on figure 1 that the energy grows quickly with respect to the distance. So the computation of shortest paths will give priority to short distances and close nodes. The optimal solution has to do the same for the same reason. So the solutions are similar and the algorithm has a good behavior.

But with the set of parameters (B), energy spent in communication towards a distant node is only up to three times greater than the energy spent in communication with a close node, within the maximum communication range of the sensor, which is 25m. This gives more opportunities for the computation of the optimal solution. The FA algorithm will still give priority to short distances. This has two bad effects. Multi-hop communication becomes more costly than direct communication, whatever the communication model, when the energy spent in transmission increases slowly with the distance between the communicating nodes. Moreover, multihop communication tends to increase the number of unintended messages in the Neighborhood model. These bad effects significantly reduce the lifetime of the network.

As a conclusion with this whole experiment, we conclude that the FA algorithm and probably all routing algorithms based on shortest paths are very sensitive to the energy parameters. The energy model has small influence on the results. It has an impact on the worst case, but not really on the average case. In the next section, we exhibit an example where the energy model has a great impact on the results.

B. LEACH Protocol

1) Description of the Protocol: In this section, we briefly describe the LEACH protocol as it can be found in [1].

The LEACH protocol is a well-known dynamic clustering protocol designed for wireless sensor networks [1], [10].

The protocol is divided in several rounds, and in each round, new cluster heads are defined and new clusters are formed. A percentage P of cluster-heads is defined before the protocol begins. Then, a node becomes a cluster-head at round r if it has not been a cluster head in the $(r \mod \frac{1}{P})$ rounds before and with a probability:

$$\frac{P}{1 - P \times (r \mod \frac{1}{P})}$$

This ensures that each node becomes a cluster head every $\frac{1}{P}$ rounds. Then, each cluster-head advertises its neighbors and each node chooses its preferred cluster-head, regarding the strength of the received signal from the different cluster-heads. Then, each node informs the chosen cluster-head that it belongs to its cluster.

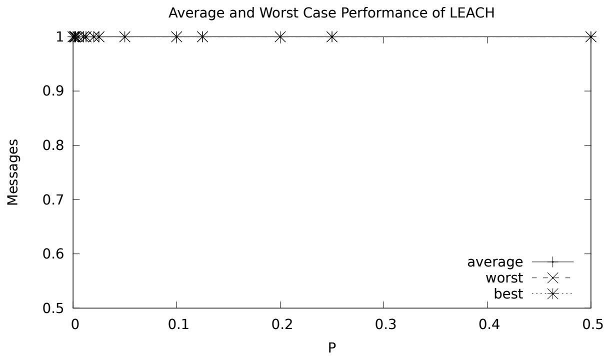

Fig. 5. Average and worst performance of LEACH relative to the optimal solution in the Sender-Receiver model

Finally, messages are sent from every node to its cluster-head and then from the cluster-head to the sink. This scheme can be extended to hierarchical clustering but we will only use this simple version.

We use similar parameters as in the original experiment. The network is composed of 50 nodes (instead of 100 in the original experiment [1]), randomly distributed in a square of 100m $\times$ 100m, with a percentage P from 0 to 0.5 of cluster-heads in each round. The length of messages is $\lambda = 2000$ bits. The other parameters are $e^T = e^R = 50$nJ/bit and $\epsilon_{\text{amp}} =$ 100pJ/bit/m^2.

As stated previously, we focus on the number of messages that are delivered. We only take the routed messages into account, not the messages dedicated to the protocol i.e. we ignore the overhead of the cluster set-up phase.

2) Results: In the original experiment, the sink is located at $(50, 200)$, i.e. 100m far from the nearest node. We did not succeed in reproducing this experiment as we could not solve the integer linear programming problem in a reasonable time. This is explained by the number of variables of the problem. The number of variables is proportional to the number of links. In the previous case, the number of links was limited due to the limited range of the nodes. In this experiment, the range is not limited so the graph is complete i.e. the number of directed edges is $n(n-1)$ if n is the number of nodes.

Moreover, as the sink is far, most messages sent by the nodes use a multi-hop path to reach the sink. The number of possible multi-hop paths is exponential relatively to the number of links. As the problem is NP-Hard, it seems that these parameters are not really adapted to a possible resolution of the optimal solution.

Therefore we placed the sink at $(90, 90)$ so that only a few messages need to use a multi-hop path to reach the sink, those that are sent by the farthest nodes. This way, we can get an optimal solution to be compared to the result given by LEACH. We ran this experiment in the Sender-Receiver model and in the Neighborhood model.

Figure 5 shows the number of messages using the Sender-Receiver model in LEACH relative to the optimal solution. We clearly see that LEACH achieves the optimal performance

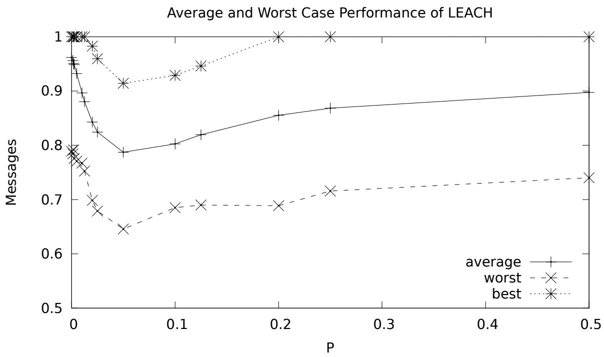

Fig. 6. Average and worst performance of LEACH relative to the optimal solution in the Neighborhood model

in any case, for any value of the P parameter.

Figure 6 is the same experiment using the Neighborhood model. In this case, the average number of messages is between 80% and 90% of the optimal solution and the worst case is around 70% of the optimal solution. This time, the difference is very significant on many points. First, the best case is not always the optimal solution. It can go down to 90% of the optimal solution for $P = 0.05$. Second, the average case and the worst case are far from the optimal solution whereas the same algorithm achieved the optimal solution in any case in the Sender-Receiver model.

These results can be explained by the clustering nature of the algorithm. In the Sender-Receiver model, the clustering achieves a good performance as expected by the original results. But clustering is not well-suited to the Neighborhood model. In this case, the optimal solution will try to minimize the number of unintended messages whereas the clustering algorithm does not take these messages into account. The assumption that the energy consumption is better if the message goes through a cluster head does not always hold in the Neighborhood model because of the unintended messages.

We can also explain the aspect of the plot. For very small P, most often, there are no cluster heads so the performance is still good. There are only a few nodes that would need a hop to improve the whole performance. When P increases up to 0.05, which means there are 2.5 cluster heads on average, the number of cluster heads in each round is rarely 0, and the number of unintended messages increases due to the retransmission of the cluster head, which decreases the performance.

Then, for $P > 0.05$, the number of cluster heads grows so much that the cluster heads tend to be closer to the nodes of their cluster. So normal nodes send fewer unintended messages to their neighbors, and the performance increases.

We conclude from this experiment that LEACH and probably all clustering algorithms are very sensitive to the communication model.

V. Conclusion

We showed that the performance of routing algorithms can be very sensitive to the communication model and the energy parameters. We provided a generalized model that can be reduced to take into account a wide range of situations regarding the type of applications or the quality of the MAC layer. We extended previous optimization problem formulation for this new model.

We evaluated the Flow Augmentation algorithm [2] and the LEACH protocol [1] and showed that the model or the parameters can have a significant impact on the results. More precisely, our analysis conclude that routing algorithms that are based on the computation of shortest paths may be sensitive to the energy parameters whereas routing algorithms that are based on clustering may be sensitive to the communication model.

These experiments could be extended to other algorithms and protocols to confirm this analysis. It could also be interesting to adapt some algorithms in order to improve their performance in any model and for any parameters.

References

[1] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*. IEEE, 2000.

[2] J. Chang and L. Tassiulas, "Maximum lifetime routing in wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 12, no. 4, pp. 609–619, 2004.

[3] P. Basu, "Effect of overhearing transmissions on energy efficiency in dense sensor networks," in *Proceedings of the Third International Symposium on Information Processing in Sensor Networks (IPSN'04)*. ACM Press, 2004, pp. 196–204.

[4] A. Sankar and Z. Liu, "Maximum lifetime routing in wireless ad-hoc networks," in *INFOCOM 2004. Twenty-Third Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 2. IEEE, 2004, pp. 1089–1097.

[5] R. Madan, Z. Luo, and S. Lall, "A distributed algorithm with linear convergence for maximum lifetime routing in wireless networks," in *Proceedings of the Allerton Conference on Communication, Control and Computing*. Citeseer, 2005, pp. 28–30.

[6] R. Madan and S. Lall, "Distributed algorithms for maximum lifetime routing in wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 5, no. 8, pp. 2185–2193, 2006.

[7] Y. Cui, Y. Xue, and K. Nahrstedt, "A utility-based distributed maximum lifetime routing algorithm for wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 55, no. 3, pp. 797–805, 2006.

[8] J. Park and S. Sahni, "An online heuristic for maximum lifetime routing in wireless sensor networks," *IEEE Transactions on Computers*, vol. 55, no. 8, pp. 1048–1056, 2006.

[9] L. Santos, M. de Oliveira Marques, and I. Bonatti, "Power-aware model for wireless sensor nodes," 2008.

[10] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.

[11] S. Schmid and R. Wattenhofer, "Algorithmic Models for Sensor Networks," in *14th International Workshop on Parallel and Distributed Real-Time Systems (WPDRTS), Island of Rhodes, Greece*, Apr. 2006.