An Energy-Efficient Geographic Routing with Location Errors in Wireless Sensor Networks

Julien Champ, Clément Saad

An Energy-Efficient Geographic Routing with Location Errors in Wireless Sensor Networks

Julien Champ
Université Montpellier 2 - LIRMM
161 rue Ada,
34392 Montpellier Cedex 5, France
champ@lirmm.fr

Clément Saad
LIA - CERI
339 chemin des Meinajaries,
84911 Avignon cedex 9, France
saad@lirmm.fr

Abstract

In wireless sensor networks, almost all geographic routing algorithms assume that sensors are accurately located. In this paper, we propose an Energy Efficient Geographic Routing algorithm (EEG-Routing). In our method, before the deployment of sensors in their environment, sensor positions are known with position error bounds which are potentially larges. According to this knowledge, it is possible to compute, before the deployment the probability that two sensors communicate. EEG-Routing introduces a new metric which defines, regarding to communication probabilities, energy consumptions and realized progress, communication costs between neighbors. EEG-Routing simultaneously optimizes two criteria: the energy consumption and the delivery rate, in networks where sensors are inaccurately located. Performances are validated by simulations which compare EEG-Routing with an energy-optimal algorithm.

1 Introduction

Recent advancements in Micro-Electro-Mechanical Systems (MEMS) and wireless communications has enabled the development of a new kind of networks: *Wireless Sensor Networks* (WSN). These networks are composed of sensors which can gather information about their environment such as temperature, gas leak detection, etc. Data is transported thanks to a multi-hop routing towards the *base station* (BS) which will process data. There are many fields where WSN can be used [1] : from forest monitoring for early fire prevention, to enemy troops movement detection.

In such networks, routing algorithms have to consider the whole characteristics of wireless sensors: each sensor is equipped with sensing, computing and communication modules. As the battery provides energy to the other modules, a routing algorithm has to select paths towards the destination that consume as less energy as possible. As a direct consequence, the network lifetime will be increased. Many solutions have been proposed to solve this problem [2, 3]. Among these solutions, some of these techniques use localization information to route messages. Almost all existing methods assume that positions are accurately known [4, 5, 6], according to GPS technology or localization techniques. In practice, localization methods do not provide accurate positions for all sensors, and equip nodes with a GPS is not reasonable (due to the number of sensors). Nevertheless, it is possible to know estimated positions with a position error bound, thanks to either some localization methods [4, 5], or according to deployement strategies.

This paper proposes a new energy-efficient geographic routing algorithm for WSN, called EEG-Routing, taking sensor position errors into account. We assume that estimated positions and position error bounds of all sensors in the network are known before deployment. These position error bounds are potentially large (up to 100% of transmission range). If positions are accurately known, two sensors are able to communicate if the euclidean distance between them is less or equal than their maximum transmission range. Due to the inacurracy of positions, it is not always possible to certify if two sensors can directly communicate but the communication probability can be calculated.

EEG-Routing progresses towards the destination according to geographic positions. In order to have a high delivery rate and a low energy consumption, each sensor forwards the message according to the knowledge of the communication probability, energy consumption and realized progress.

This paper is organized as follows: in section 2, we present the hypothesis and the modelisation of the routing problem, then in section 3, we propose a metric to define communication costs. EEG-Routing algorithm is presented in section 4, and section 5 is dedicated to simulation results. Section 6 ends the paper.

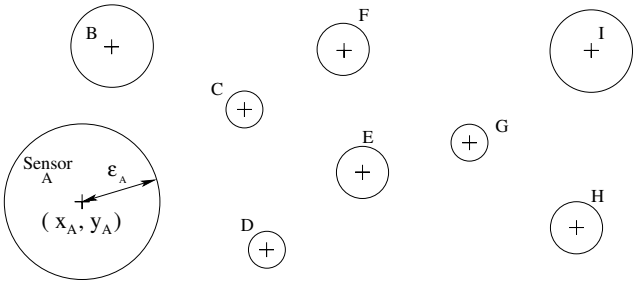

Figure 1. Network representation known before its deployment

2 Hypothesis and Modelisation

It is known that in some applications, sensors can be mobile, we considers here that sensors are static. We also assume that all the sensors have the same maximum transmission range r. If the distance between two sensors is less or equal than r, they can communicate. However, all sensors can adjust their transmission range in order to transmit on less distance and so reduce their energy consumption. We consider that there is only one base station in the network, and its position is known by all sensors.

Before deployment, for example thanks to helicopter, it is possible to know the final positions to all sensors in the network. The position accuracy will depend on velocity, altitude of the helicopter, and other parameters.

We assign to each sensor u an estimated position (x_u, y_u), the x and y coordinates of u, and a position error bound ϵ_u. The real position of the node u is inside the disk centered in (x_u, y_u), of radius ϵ_u. We consider that the real position of a sensor can be anywhere inside the disk with the same probability (i.e. there exist a uniform distribution of the position in the disk). The distance between two nodes u and v, denoted d_{uv}, is the euclidean one. Figure 1 is an example of the network representation before the deployement of sensors by helicopter.

We represent a wireless sensor network by an oriented graph $G = (V, E)$ where V is the node set (the sensors) and $E \subseteq V^2$ the arc set such that the communication probability is greater than 0. Two nodes are neighbors if they can directly communicate. The set E is defined as follows:

$$E = \{(u, v) \in V^2 \mid |d_{uv} - \epsilon_u - \epsilon_v| \leq r\}$$

Due to this definition, the graph G is a symetric one. We have chosen an oriented graph because next, we will associate a cost for each arc.

3 Definition of Metric for Arc Cost

This section proposes a new metric to associate a cost to each arc of the graph. Let A and B be two sensors, (x_A, y_A), (x_B, y_B) their estimated positions and ϵ_A, ϵ_B their position error bounds. The cost of an arc (A, B) is defined thanks to the three following criteria: the probability that nodes A and B can communicate, the energy consumption [1] and the progress realized when sensor A sends the message to B. We define the realized progress as the difference between the euclidean distance from the sender to the base station and the distance from the receiver to the base station. First, we will explain how to estimate the probability p_{AB} of communication between nodes A and B, and the ratio R_{AB} between the energy consumption and the realized progress when node A sends the message to node B.

3.1. Communication probability calculation

In this section, we define a function to estimate the communication probability between two sensors A and B, according to sensor estimated positions, position error bounds and the transmission range r.

To compute p_{AB}, we differentiate three cases: first, two sensors are located with exact positions (i.e. $\epsilon_A = \epsilon_B = 0$). Second, only one sensor is located with an estimated position and the other node is exactly located (i.e. $\epsilon_A \neq 0 \; and \; \epsilon_B = 0$). In the last case, the two sensors are located with estimated positions (i.e. $\epsilon_A \neq 0 \; and \; \epsilon_B \neq 0$).

The first case is the simplest. If nodes are exactly located, the probability is obtained as follows:

$$p_{AB} = \begin{cases} 1 \, , \; if \; d_{AB} \leq r \\ 0 \, , \; otherwise \end{cases} \quad (1)$$

In other words, when the euclidean distance between A and B is less or equal than the maximal transmission range r the probability p_{AB}, the two nodes can communicate.

In the second case, p_{AB} is computed as being an area ratio: let S be the area defined by the intersection of the disks centered respectively in A and B, of radius respectively equals to ϵ_A and r:

$$p_{AB} = \frac{S}{\pi \times \epsilon_A^2} \quad (2)$$

The last case is the most difficult. It is illustrated in figure 2 :

- The S_1 area contains possible positions for sensor A when A and B cannot communicate.
- The S_3 area contains possible positions for sensor A when A and B can communicate.

[1] due to the wireless communication

- In the S_2 area, it is not possible to guarantee if sensors A and B are able to communicate. The probability of communication in S_2 has to be estimated.

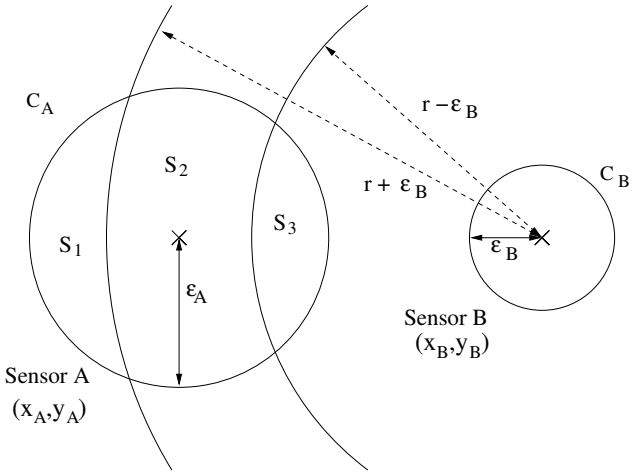

Figure 2. Sensor B sends a message to sensor A, ϵ_A and $\epsilon_B > 0$

Let p_1, p_2 and p_3, communication probabilities when sensor A is respectively located in S_1,S_2 and S_3. The communication probability p_{AB} between two sensors A and B depends on areas S_1,S_2,S_3, and probabilities p_1, p_2 and p_3. Probability p_{AB} is obtained by the following formula:

$$p_{AB} = \frac{S_1 * p_1 + S_2 * p_2 + S_3 * p_3}{\pi * \epsilon_A^2} \tag{3}$$

When sensor A belongs to area S_3, sensors A and B are sure to be able to communicate, so $p_3 = 1$. Conversely, if A belongs to S_1 , sensors A and B are not able to communicate, so $p_1 = 0$. The calculation probability p_{AB} is simplified as follows:

$$p_{AB} = \frac{S_2 * p_2 + S_3}{\pi * \epsilon_A^2} \tag{4}$$

The p_2 probability calculation is more difficult. Due to lack of place, the reader is invited to read [7] for more details. The next section focus on the calculation of the ratio between energy consumption and the realized progress.

3.2. Ratio between energy consumption and realized progress

In this section, our goal is to propose a normalized value R_{AB} between 0 and 1, as a function of energy consumption and progress realized when sensor A sends a message to B.

We use the most commonly used energy consumption model presented in [8]. The energy consumption function J, as a function of transmission range d, is defined as follows:

$$J(d) = d^a + c \tag{5}$$

As authors in [8], we consider this model with $a = 4$ and $c = 2 \times 10^8$.

Optimal transmission range : For this given energy consumption model, there exists a transmission range where the ratio between enery consumption and this transmission range is optimal. The ratio E is calculated as follows :

$$E(d) = \frac{J(d)}{d} = \frac{d^4 + 2 \times 10^8}{d} \tag{6}$$

To calculate the optimal transmission range, we have to calculate the E(d) derivative function denoted E'(d):

$$E'(d) = 3d^2 - \frac{2 \times 10^8}{d^2} \tag{7}$$

We obtain the optimal transmission range d_{opt} when E'(d)=0 :

$$d_{opt} = \sqrt[4]{\frac{2 \times 10^8}{3}} \approx 90.36 \tag{8}$$

We are going to use this optimal transmission range in order to normalize ratio R between 0 and 1.

The realized progress is defined as the euclidean distance between the emitter node and the base station minus the distance between the receiver node and the base station. When sensor A sends a message to sensor B, then the realized progress $prog_{AB}$ is :

$$prog_{AB} = d_{A,BS} - d_{B,BS} \quad \textit{, with BS the base station} \tag{9}$$

For a given energetic model there exist a ratio which is the best trade-off between energy consumption and realized progress. This optimal ratio is obtained as follows:

$$E(d_{opt}) = \frac{J(d_{opt})}{d_{opt}} \tag{10}$$

The ratio corresponding to (A, B) arc is given by:

$$\frac{J(d_{AB})}{prog_{AB}} \tag{11}$$

We want to penalyze arcs not having a ratio close to the optimal one, and so we define the ratio R_{AB} belonging to $[0, 1]$ as follows:

$$R_{AB} = \frac{\frac{E(d_{opt})}{J(d_{AB})} + 1}{\frac{prog_{AB}}{2}} = \frac{prog_{AB} \times E(d_{opt}) + J(d_{AB})}{2J(d_{AB})} \quad (12)$$

Given that $J(d_{AB}) = J(d_{BA})$, but as $prog_{AB} = -prog_{BA}$, so $R_{AB} \neq R_{BA}$ for arcs (A, B) and (B, A).

3.3. Arc cost calculation

The previous sections explain how to calculate p_{AB} and R_{AB}, then we have to combine these values to define the cost of arc (A, B). The metric penalizes an arc (A, B), by giving it a high cost when it does not present a good trade-off between the probability p_{AB} and the ratio R_{AB}. For all arc $(A, B) \in E$, the cost of the arc (A, B), denoted C_{AB}, is given by the following formula:

$$C_{AB} = 1 - [\alpha\, p_{AB} + (1-\alpha) R_{AB}] \quad \textit{where } \alpha \in [0, 1] \quad (13)$$

Thanks to α, it is possible to give more importance to p_{AB} probability, or R_{AB} ratio

As the new metric is defined, the shortest path between a sensor and the base station can be computed thanks, to Dijkstra Shortest Path algorithm for example.

4 The Routing Algorithm

Before deployment, a computer determines for each sensor, the least path cost to reach the base station for its possible neighbors. Then each sensor will store a table, denoted Tab_Costs, associating to each neighbor the cost to reach destination.

After deployment, each sensor learns its neighbourhood after an exchange of $HELLO$ messages. A sensor deletes in its table Tab_Costs nodes which finally do not belong to its neighbourhood. The main idea of our algorithm is the following: when a sensor wants to send a message, thanks to its knowledge, it sends the message to the neighbor having the least cost in its table Tab_Costs. Each sent message contains the following data:

- the position of the sensor which detected an event,
- the cost of the backup path bkp (a *backup path* is computed as being the second least value in its table Tab_Costs),
- a message identifiant id,
- detected event information (temperature, gas...).

When v receives a message sent by u, sensor v deduces that it is the best choice of u. It updates its table Tab_Costs, by modifying the associated cost of sender u as follows: the cost of the backup path bkp_u, included in the message, plus the cost C_{uv}. Next, v will select the neighbor having the least cost in its table Tab_Costs, and forwards the message including its own new backup path cost. The main part of this method is presented in algorithm 1.

Algorithm 1: When sensor v receives message Msg sent by u

```
/* Msg= < Id, Snd, B_Snd > */
/* Id the ID-message, Snd the sender identifiant and B_Snd the
backup path cost */
/* bestNeighbor the current best selectionned neighbor to send the
message towards the BS */
/* forbid(neighbor) Forbid the sensor to send the message to
neighbor */
/* store(Msg.id) Store the ID-message */
/* received(Msg.id) returns False when Msg is received for the
first time, True otherwise */
/* bestPath() returns the neighbor having the best path among not
forbidden neighbors, -1 otherwise */
/* backupPath() returns the second best path among not forbidden
neighbors, -1 otherwise */

if received(Msg.id) = True and u ≠ bestNeighbor then
    forbid(bestNeighbor) // for Loop Free
if bkp_Snd ≠ −1 then
    Tab_Costs[u] = bkp_Snd + C_uv
    store(Msg)
else
    forbid(u) // There are no paths towards destination passing by u
bestNeighbor = bestPath()
Msg.bkp_v = backupPath()
if bestNeighbor ≠ −1 then
    send(bestNeighbor, Msg)
else
    Give up
```

Note that, the selected path is not the same as one goes along the message progress towards the base station, because each sensor will take its decision according to its own knowledge. This kind of behaviour can lead to loops in the routing algorithm.

Two main properties of EEG-Routing algorithm are that it's a loop-free and fault tolerant algorithm:

Loop-free: In order to avoid message loops, we take the following decision: if sensor u sent a message to sensor v, and if the message comes back to u by another sensor than v, this means that a loop happened. Thus, if the sensor u forbids itself to send again the message to v, it prevents this loop from happening another time. Therefore, the message ID must be stored during a certain amount of time.

Fault tolerant: If a sensor periodically sends an I_AM_ALIVE message, it is possible to detect failures. When a sensor break down, its neighors detect this event after some time (i.e. no I_AM_ALIVE message received from this sensor) and update data about this sensor in their Tab_Costs tables. This detection is not propagated to the whole network. Thus, sensor failures are managed.

5 Simulations

5.1. Simulation environment

In order to measure the interest of EEG-Routing, we have implemented this algorithm in a simulator we developed. In our simulations, the MAC layer is considered as being ideal (lossless and collision-free). Position error bounds are normalized to the radio range. For example, a node with a 50% position error bound means that the distance between its estimated and real positions is less or equal than half of transmission range.

Generated network toplogies contain 100 sensors in a 1200×1200 area. So as to have a density between 6 and 20, the maximum transmission range r is adjusted. For each node $u \in V$ (V is the set of nodes), we randomly pick with an uniform distribution, following data :

- u estimated position (x_u, y_u) inside the 1200×1200 square,
- u position error bound ϵ_u between 0 and 100%,
- u real position inside the disk centered in (x_u, y_u) of radius ϵ_u.

The base station position is also randomly picked , but without position error. For each density, 100 topologies are generated. During scenarii, each sensor detects an event. In this case, it sends a message towards the base station. As EEG-Routing is based on different hypothesis, like the knowledge of all node positions with position error bounds, it is not compared to other geographic routing algorithms. To evaluate EEG-Routing, its results are compared with the ones obtained with an energy optimal algorithm. In the energy optimal algorithm, communication links are known. The least cost path is computed thanks to the global knowledge of the network. Path costs are calculated regarding to energy consumption. Thanks to this centralized algorithm, the delivery rate and the energy consumption are optimal.

5.2. Simulation results

The graphs represented in figure 3 show the impact of α parameter: the first graph analyzes the impact on the delivery rate, and the second graph the impact on the energy consumption.

In high density networks, whatever value of α the delivery rate is close to 100%. In low density networks, it is preferable to use a low value for α, giving more importance to progress. Thus, messages are mainly sent towards neighbor nodes which are close to the destination regarding to senders. It will prevent the appearance of loops and thus it will lead to a better delivery rate.

Whatever value of density, the higher α is the better the energy consumption is. When we increase the value of α, we give more consideration to the probability of communication. Therefore, EEG-Routing increases its chances to use arcs which exists. However, it is necessary to avoid a value near 1 for α to take realized progress and energy consumption into account.

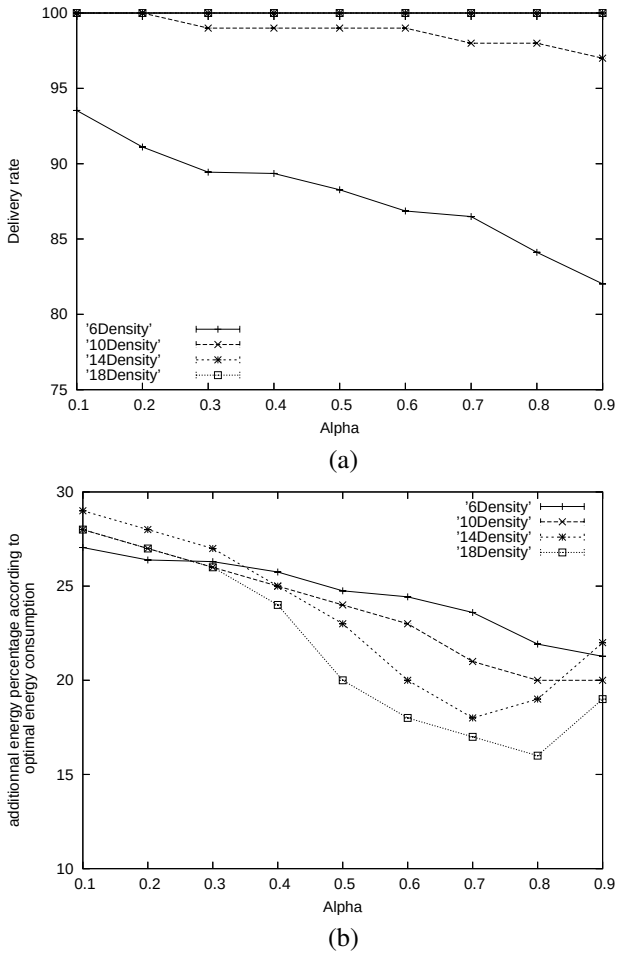

Figure 3. Impact of α parameter on : (a) delivery rate (b) energy consumption

The table 1 indicates the value to choose for parameter α to obtain, when it is possible, more than 99% of message delivery rate while the energy consumption is minimal.

We can see that, when density increases, it is preferable to choose a higher value for constant α.

Density	6	8	10	12	14	16	18	20
α	0.1	0.3	0.6	0.7	0.7	0.7	0.8	0.8
Delivery rate	94	99	99	99	100	100	100	100

Table 1. Best value for parameter alpha according to density

In the graph represented in figure 4, the network density varies and we focus on the additional energy percentage according to optimal energy consumption. For each density, we choose the best α parameter regarding to the table 1. We can see that our method is scalable, energy consumption decreases when density increases. In low density networks (i.e. density equals to 6) our algorithm consumes less than 30% of additional energy according to optimal energy consumption. In high density networks, the energy consumption is less than 20%.

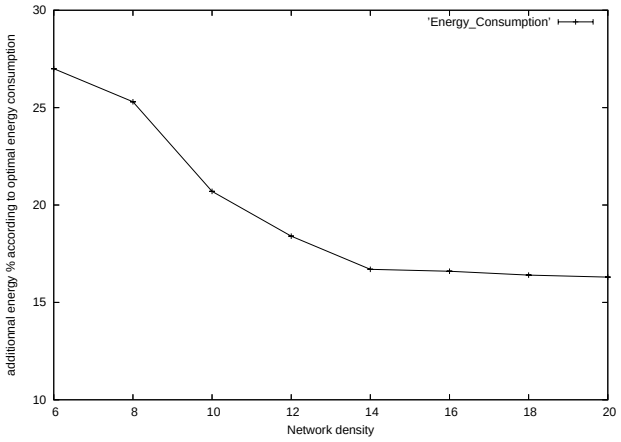

Figure 4. Impact of network density on energy expenditure.

6 Conclusion

This paper presents EEG-Routing, a new geographic routing algorithm for WSN, based on estimated positions with position error bounds. Due to position errors, communications between sensors are known with probabilities. EEG-Routing sends messages along paths having the best trade-off between communication probability, progress and energy consumption.

Some improvements can be made to this method. To increase network's lifetime, if a sensor knows the energy level of its neighbors, it can penalize the ones which have a low energy level to avoid to use them as relay nodes when routing messages. Thanks to this improvement, the first sensor break down (i.e. due to lack of energy) will happen later. The loop management can be improved: instead of forbiding the usage of some arcs, we should penalize them. This improvement may permit to have a delivery rate equals to 100%, while energy consumption increases (the worst case would be the full depth first search of the graph). Finally, as future work, we want to consider the performances of EEG-Routing in a more realistic environment using real devices to validate this method by experimentations.

Acknowledgements: Authors wish to thank Anne-Elisabeth Baert, Jean-Claude König and Jérôme Palaysi for their precious advices and constructive comments.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 2002.

[2] I. Stojmenovic and X. Lin. Power-aware localized routing in wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 2001.

[3] J. Kuruvila, A. Nayak, and I. Stojmenovic. Progress based localized power and cost aware routing algorithms for ad hoc and sensor wireless networks. *International Journal of Distributed Sensor Networks*, 2006.

[4] C. Saad, A. Benslimane, and J-C. König. At-dist : A distributed method for localization with high accuracy in sensor networks. *Special Issue on "Wireless Ad Hoc and Sensor Networks" in the international journal Studia Informatica Universalis (To Appear)*, 2007.

[5] C. Saad, A. Benslimane, J-C. König, and J. Turbert. At-free : A preliminary method for localization techniques in sensor networks. *7th International Conference on New Technologies of Distributed Systems*, 2007.

[6] D. Niculescu and B. Nath. Ad hoc positioning system. *in Proceedings of GLOBECOM, San Antonio*, 2001.

[7] J. Champ. *Routage dans les réseaux Ad hoc*. Master Thesis, LIRMM, June 2007.

[8] V. Rodoplu and T.H. Meng. Minimum energy mobile wireless networks. In *Proc. of IEEE International Conference on Communications (ICC)*, 1998.