



**HAL**  
open science

# Energy-based Clustering for Wireless Sensor Network Lifetime Optimization

Tony Ducrocq, Nathalie Mitton, Michaël Hauspie

► **To cite this version:**

Tony Ducrocq, Nathalie Mitton, Michaël Hauspie. Energy-based Clustering for Wireless Sensor Network Lifetime Optimization. WCNC - Wireless Communications and Networking Conference - 2013, Apr 2013, Shanghai, China. hal-00767690

**HAL Id: hal-00767690**

**<https://inria.hal.science/hal-00767690>**

Submitted on 12 Jun 2013

**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.

# Energy-based Clustering for Wireless Sensor Network Lifetime Optimization

Tony Ducrocq  
Inria Lille - Nord Europe  
tony.ducrocq@inria.fr

Nathalie Mitton  
Inria Lille - Nord Europe  
nathalie.mitton@inria.fr

Michaël Hauspie  
Université Lille1  
michael.hauspie@lifl.fr

**Abstract**—Clustering in wireless sensor networks is an efficient way to structure and organize the network. It aims to identify a subset of nodes within the network and bind it a leader (*i.e.* cluster-head). This latter becomes in charge of specific additional tasks like gathering data from all nodes in its cluster and sending them by using a longer range communication to a sink. As a consequence, a *cluster-head* exhausts its battery more quickly than regular nodes. In this paper, we present **BLAC**, a novel Battery-Level Aware Clustering family of schemes. **BLAC** considers the battery-level combined with another metric to elect the cluster-head. It comes in four variants. The *cluster-head* role is taken alternately by each node to balance energy consumption. Due to the local nature of the algorithms, keeping the network stable is easier. **BLAC** aims to maximize the time with all nodes alive to satisfy application requirements. Simulation results show that **BLAC** improves the full network lifetime 3-time more than traditional clustering schemes by balancing energy consumption over nodes and still delivering high data percentage.

## I. INTRODUCTION

Multi-hop wireless sensor networks (MWN) consist of sets of mobile wireless nodes without the support of any pre-existing fixed infrastructure. Such large scale wireless sensor networks offer great application perspectives. Sensors are tiny devices with hardware constraints (low memory storage, low computational resources) that rely on battery. Sensor networks thus require energy-efficient algorithms to make them work properly in a way that suits their hardware features and application requirements. In this paper, we focus on a given application defined by the *BinThatThinks*<sup>1</sup> project that aims to ease the collect and recycling of waste and reduce its cost through the use of wireless sensors placed on dustbins. Dustbins are also equipped with GPRS chips for long range communications.

The goal of this paper is to propose a novel clustering algorithm for wireless sensor networks in which each sensor node sends its data to its *cluster-head* (potentially through multi-hop paths) based on the context of the *BinThatThinks* project. In this framework, *cluster-heads* collect data from all sensors in their cluster and send them through their GPRS link. Since activating the GPRS consumes more energy than peer-to-peer communications, each node should take the *cluster-head* role in turn in order to allow the network to be operational as long as possible without too much communication overhead and structure modifications. Data aggregation is performed at

every hop toward *cluster-head*. The clustering scheme should also provide a good trade-off between the number of clusters (the more clusters, the more GPRS activation) and the size of the cluster radius (the bigger radius, the more peer-to-peer communications to reach the cluster-head). The network needs to last as long as possible without any dead node. Energy should be equally distributed over nodes. In this paper, we present **BLAC**, a novel family of **B**attery-**L**evel **A**ware **C**lustering protocols.

To the best of our knowledge, **BLAC** is the very first distributed clustering algorithm providing non-overlapping multi-hop clusters with energy concerns. Solutions from the literature mainly propose clustering schemes that either do not balance energy consumption over nodes or provide non-suitable clusters for our application (overlapping or one-hop clusters). **BLAC** considers the battery-level of nodes combined with another metric (density and degree) to elect the *cluster-head*. Yet, nodes naturally change roles over time based on node energy level but in a limited way in order to provide stability to the structure. **BLAC** comes in four variants: **BLAC-bg** combines battery level and node degree, **BLAC-bs** uses the battery level and node density. **BLAC-rg** and **BLAC-rs** run in two steps. They first apply a graph reduction before computing clustering. Each of these variants presents specific features that make them more suitable than others under different conditions. Simulation results show that **BLAC** extends the lifetime of the first dying node up to 300% compared to literature works, by balancing energy consumption over nodes and allowing a better delivery ratio.

The rest of the paper is organized as follows: Section II presents relative works for clustering on wireless sensor networks. The context and the application targeted by **BLAC** are introduced in Section III. Section IV describes previous algorithms used by **BLAC** for self-content purposes. Section V details **BLAC**. Simulation results are given in Section VI. We finally conclude in Section VII.

## II. RELATED WORK

Literature proposes several techniques for cluster formation and *cluster-head* selection. All solutions aim to identify a subset of nodes within the network and bind it a leader. First solutions such as LCA [1] and HCC [6] present a similar clustering structure but differ in the metric used. Each node chooses its parent in its neighborhood such as its parent metric is higher than its own one. If a node has the highest metric value in its neighborhood it becomes *cluster-head*. LCA uses as metric the unique Id of a node and HCC the degree then the Node identifier

This work is partially supported by CPER Nord-Pas-de-Calais/FEDER Campus Intelligence Ambiante and ANR ECOTECH BinThatThinks.

<sup>1</sup><http://binthatthink.inria.fr>

(*Id*) to break ties. LCA and HCC create one-hop overlapping clusters which are quite small. A cluster is constituted of a cluster-head and all its one-neighbors. Two *cluster-heads* can not be neighbors and a non-*cluster-head* node belongs to at least one *cluster*. Overlapping clusters are subject to chain reactions when changes occur. Such algorithms are thus not suitable to large dynamic networks. In addition, they do not have energy concerns.

WCA [3] considers multiple parameters like positions, mobility and energy. It computes the best weighted sum for a given application and uses this metric in order to elect *cluster-heads*. WCA is one of the first clustering algorithms taking energy into account in its construction. Nevertheless, it produces one-hop clusters, which is not stable. In addition, the algorithm is centralized which makes it hardly scalable.

LEACH [7] uses energy as a priority in the clustering design. Each node becomes *cluster-head* based on a predefined probability  $p$  defined based on the number of expected clusters and on whether it has already been *cluster-head* in previous rounds. A node can be elected as *cluster-head* even if its remaining energy is low. This node can then die rapidly, triggering a new *cluster-head* selection and related traffic. Once *cluster-heads* are advertised, nodes attach to the *cluster-head* which requires the least energy to be reached, creating one-hop clusters. Then, every *cluster-head* synchronizes the nodes belonging to its cluster and assign them a time slot for their data transmissions. The percentage of *cluster-heads* needs to be tuned off-line to fix  $p$  and does not dynamically adapt to underlying topologies. Also, by construction, there is a probability for nodes to have no *cluster-head* in their neighborhood, leading to data losses.

Other propositions exist in the literature but most of them are centralized [4]. Some focus on pre-deployment analysis [13], [8] and thus are not scalable. Very few of them are energy-efficient [15], [2] and even if it is the case, they require additional information [9] which is not necessarily available at nodes (like GPS position), require expensive data exchange or need a declared sink [5].

In this paper, we focus on distributed multi-hop non-overlapping energy-efficient clusters with no predefined size in order to match the underlying network topology and to be reliable to small topology changes. DDR [11] and Density-based algorithms [10] are distributed algorithms proposing multi-hop non-overlapping clusters. They run a similar clustering algorithm and mainly differ in the metric used. DDR [11] uses the degree as metric while Density-based [10] introduces a new metric called *density*. More details are given in Section IV-B. Although efficient in terms of reliability, these two propositions do not consider energy constraints. **BLAC** is thus based on similar algorithm but differs in the metric used, making the cluster organization dynamic regarding to evolution of energy consumption. This maintains the network available as a whole as long as possible.

Unlike solutions from literature, **BLAC** builds dynamic energy-efficient multi-hop clusters in a distributed way. Its main goal is to extend network lifetime (measured as the lifetime of the first node running out of energy).

### III. PROBLEM STATEMENT, CONTEXT AND MOTIVATION

This paper is conforming the *BinThatThinks* project. The project aims to ease the collect and recycling of waste and reduce its cost. Dustbins are equipped with (i) a wireless module to allow wireless peer to peer communications and (ii) a GPRS modules for longer range communications with the base station. Dustbins then periodically send information about their location, their filling level (does this garbage need to be emptied? how many trucks are needed?), accuracy of the recycling, adequate content, etc. All these data are processed by the garbage truck to dynamically optimize the collect path and by the operators to reduce useless manipulations of empty or dangerous dustbins. Since GPRS consumes much more power than a peer to peer communication chip, it is not wise to use it on all nodes all the time.

The idea defended in *BinThatThinks* is as follows. In order to reduce power consumption for each node, a clustering organization is run over the network. Each node sends its data to its *cluster head*. Once all data are gathered, the *cluster head* aggregates them and sends them to a base station or the garbage truck by using its GPRS module. By doing so, only the *cluster head* activates its GPRS module and drains more energy (mainly because of the use of GPRS module). **BLAC** proposes a dynamic energy-efficient trade-off between the size and the number of clusters.

### IV. NOTATIONS AND PRELIMINARIES

**BLAC** relies on previous works from the literature. For the sake of self-content and clarity, this section introduces other works on which **BLAC** rely and related notations.

#### A. Generic notations

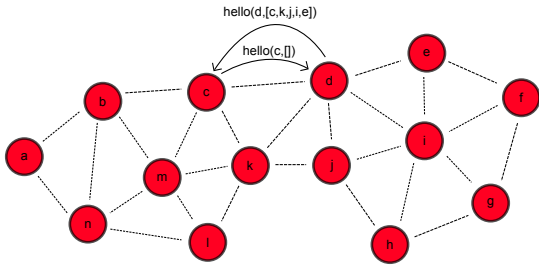
We model a wireless sensor network as a graph  $G = (V, E)$  where  $V$  is the set of sensors and  $E$  is the set of wireless links  $uv$  between each pair of sensors  $u$  and  $v$  which are in radio range of each other. We note  $\mathcal{N}(u)$  the neighborhood of a node  $u$  i.e.  $\mathcal{N}(u) = \{v \mid uv \in E\}$  and  $\delta(u) = |\mathcal{N}(u)|$  the degree of node  $u$ . In **BLAC**, clusters are formed through a tree construction in which each node has a parent node. We denote as  $P(u)$  the parent of node  $u$  in its cluster tree. We note  $\mathcal{H}(u)$  the *cluster-head* of the cluster for  $u$ .

#### B. Density metric

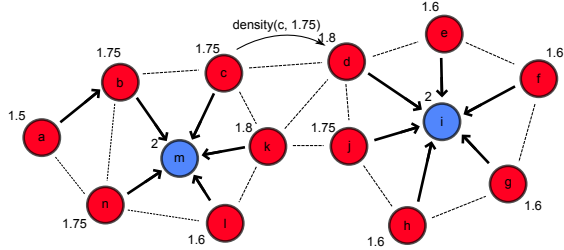
In [10], clusters are built as follows. Every node  $u$  elects its parent within its neighborhood as the node with the highest density. If  $u$  is the node with highest density in its neighborhood, it elects itself as its parent and becomes a *cluster-head*. Ties are broken by selecting the node with lowest *Id*. If a node  $v$  is neighbor of two *cluster-heads*, it advertises them and the *cluster-head* with lowest density elects node  $v$  as its parent. This ensures a minimal 3-hop distance between *cluster-heads*. The density  $\rho$  is defined as the ratio of the number of links between and to neighbors of  $u$  over the degree of  $u$ :

$$\rho(u) = \frac{|\{v, w \in E \mid v \in \{u, \mathcal{N}(u)\}, w \in \mathcal{N}(u)\}|}{\delta(u)}$$

Figure 1(b) shows an example of a network on which we compute the density of nodes. For instance node  $a$  degree's,



(a) Each node sends a *hello* message to its neighbors containing its Id. A second *hello* message is sent with list of neighbors, then each node knows its 2-hop neighborhood.



(b) Nodes can compute their *density* according to informations they gathered, then they send it to their neighbors. Finally they attach to the node with the highest density in their neighborhood. Nodes with highest density withing their neighborhood become *cluster-heads*.

Fig. 1. Clusters creation.

$\delta(a)$  is equal to 2 and the number of links between neighbors of  $a$  is 1. Then  $\rho(a) = \frac{2+1}{2} = 1.5$ .

### C. Topology reduction

Two variants of **BLAC** (**BLAC-rg** and **BLAC-rs**) run in two steps. In the first step, a graph reduction that takes account of node energy level is computed as in [12]. This is performed by applying a Relative Neighborhood Graph (RNG) [14]. RNG consists of logically removing the edge with the weakest weight in every triangle of the graph. Authors of [12] introduce a new metric called *power factor*. The *Power factor* of a link is based on the energy level of nodes at each end of the link. If both nodes are in normal battery state, *power factor* is 0, if only one node is in critical battery state *power factor* is 1 and if both nodes are in critical battery state *power factor* is 2. More common metrics for **RNG**, *i.e.* the Received Signal Strength Indicator (RSSI) and node ids are used to discriminate nodes.

Fig. 4(a) illustrates the result of a battery-level aware RNG computation. Note that most links between two low energy nodes are discarded from the resulting graph. For instance, the link between nodes  $n$  and  $m$  has been removed, as well as the one between  $a$  and  $n$ . The link between  $j$  and  $h$  is kept because there is no link between  $j$  and a neighbor of  $h$  with a better power factor than link  $jh$ . Nodes with high level of energy are preferred for communications.

## V. CONTRIBUTION

In order to maximize the lifetime of the network we introduce **BLAC**. **BLAC** comes in 4 **distributed** and **local** variants that **not need large scale modifications** when local changes appear. **BLAC** aims to keep as many nodes alive as long as possible. The role of *cluster head* is played by every

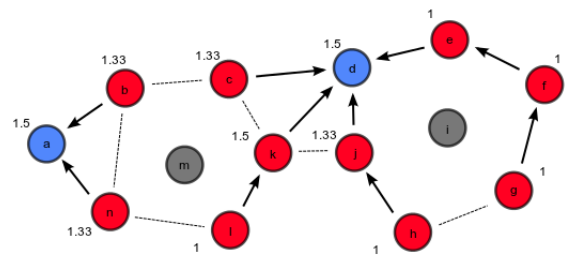


Fig. 2. Battery drain: case of density. *Cluster-heads* die faster than other nodes, creating holes in the network.

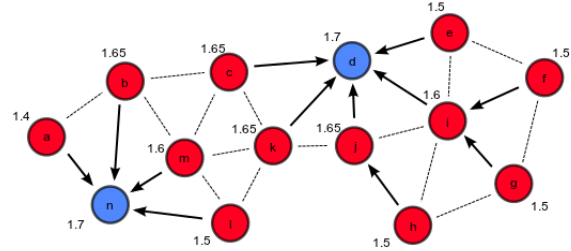


Fig. 3. battery drain : case of **BLAC-b\***. After *cluster-heads* drain their batteries faster than other nodes, *cluster-heads* roles change.

node in turns in order to **balance the energy consumption**. All variants are similar to Density-based [10] but the metrics used differ.

**BLAC** combines the *remaining energy*  $B(u)$  with another metric. We define the *remaining energy* of node  $u$  as :

$$B(u) = \lfloor \frac{batt(u) \cdot 10}{battcap} \rfloor$$

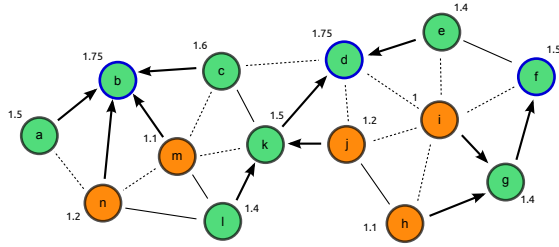
where  $battcap$  is the initial capacity of the node battery (same for every node) and  $batt(u)$  is the current battery level of node  $u$ . Then the remaining power  $B(u)$  is an integer between 0 and 10 to limit frequent changes in the metric which would result in a non stable cluster hierarchy.

Four variants of **BLAC** are **BLAC-bg** and **BLAC-bs** from one hand (detailed in Section V-A), **BLAC-rg** and **BLAC-rs** from the other hand (**BLAC-rg** and **BLAC-rs** in Section V-B).

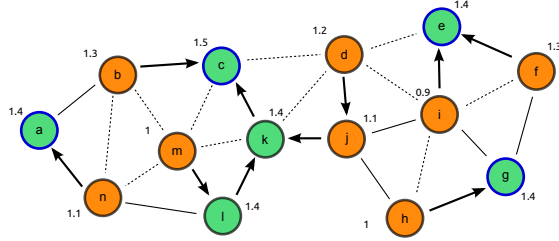
### A. **BLAC-b\*** algorithms

**BLAC-bg** and **BLAC-bs** apply the same algorithm but differ in the metric they use. We first detail the metrics they use.

1) *Degree and energy: BLAC-bg for Battery-Level Aware Clustering - Battery deGree* is based on node degree. Algorithm 1 is run over all nodes with  $h(u) = \delta(u)$  (Section IV-A). This variant uses a 1-hop neighborhood to build the network, so it stabilizes quickly. However the clustering structure built is sensitive to node apparition and failure as shown in [10] since any single change has a direct impact on neighbors and so on degree.



(a) Initial step.



(b) After a while, underlying topology changes according to energy situation. *Cluster-heads* changes based on new topology, energy is balanced.

Fig. 4. battery drain : case of **BLAC-r\***. Nodes with low energy level are displayed in orange and nodes with high energy level are displayed in green. A battery-level based **RNG** is applied removing links between low energy nodes. Arrows represent the choice of parent, dash links are links removed at the RNG step.

2) *Density and energy*: **BLAC-bs** for **Battery Level Aware Clustering - Battery denSity** uses the density  $\rho(u)$  as described in Section IV-B. Algo. 1 is run with  $h(u) = \rho(u)$ . This variant computes the clustering structure with 2-hop information but the stability is improved because a single node has less impact on its neighbors.

3) *BLAC-b\**: Once the metric is computed, **BLAC** runs Algorithm 1 with  $h(u)$  being either the degree ( $\delta(u)$ ) for **BLAC-bg** variant or the density ( $\rho(u)$ ) for **BLAC-bs** variant and by considering the battery level jointly with that metric ( $g(u) = B(u)$ ).

---

**Algorithm 1: BLAC-\* algorithm run at each node  $u$ .**

---

```

1  $C_h \leftarrow 0$ ;
2  $Id_{min} \leftarrow +\infty$ ;
3 if RNG Variants then
4    $\mathcal{R}(u) \leftarrow RNG(\mathcal{N}(u))$ ,  $g(u) = 1$ ;
5 else
6    $\mathcal{R}(u) \leftarrow \mathcal{N}(u)$ ,  $g(u) = B(u)$ ;
7 for  $i \in \mathcal{R}(u)$  do
8   if  $C_h < h(i) \times g(i) \vee (C_h = h(i) \times g(i) \wedge Id(i) < Id_{min})$ 
9     then
10     $P(u) \leftarrow i$ ,  $C_h \leftarrow h(i) \times g(i)$ ,  $Id_{min} \leftarrow Id(i)$ 
11 if  $h(u) \times g(u) > C_h \vee (h(u) \times g(u) = C_h \wedge Id(u) > Id_{min})$  then
12   /*  $u$  becomes cluster-head */
13    $\mathcal{H}(u) \leftarrow u$ ;
```

---

Algo. 1 runs at each node as follows : each node sends to its neighbors a *hello* message with its *Id*. Knowing its neighborhood, a node sends a new *hello* message with its *Id* and its neighbors *Ids* (Fig. 1(a)). With this information, nodes can compute their *density* and send it to their neighbors (Fig. 1(b)). Finally nodes can elect their parent or decide to

elect themselves as a *cluster-head* if they have the highest metric (Fig. 1(b): nodes  $m$  and  $i$  become *cluster-heads*). Note that clusters are thus built through a tree construction :  $a$  elects  $b$  that elects  $m$  that elects itself as a *cluster-head*. This same tree is then used to route data towards the *cluster-head*.

At the beginning all nodes have the same battery level so the algorithms act like Density-based or DDR respectively. Battery starts to drain differently on each node regarding the neighborhood of the node, its activity and whether it is a *cluster-head* or not. For instance on Figure 3  $B(m)$  will decrease slower than  $B(n)$ , then  $m$  will leave its *cluster-head* role to  $n$ .

As Fig. 2 shows, plain *Density-based* clustering scheme will eventually create holes. First, nodes  $m$  and  $l$  will die because *cluster-heads* consume more energy than regular nodes. After some reorganization  $a$  and  $d$  are the new *cluster-heads* and they will finally die, leaving holes in the network, and nodes, which have never been *cluster-heads*, with a lot of energy.

With **BLAC** (Fig. 3), instead of dying, nodes  $m$  and  $l$  will leave their *cluster-heads* roles before their energy level run to low. In their turns, the new *cluster-heads*  $n$  and  $d$  will leave their roles to  $b$  and  $l$ , balancing the energy consumption of the whole network.

## B. BLAC-r\*

**Battery-Level Aware Clustering - RNG deGree (BLAC-rg)** and **Battery-Level Aware Clustering - RNG density (BLAC-rs)** variants are variations of the first and the second ones. The main difference is that Algorithm 1 runs in two steps. Before computing its metric (degree or density), a relative neighborhood graph [14] is computed in order to keep only an interesting subset of nodes (Lines 4-5 in Algorithm 1). This allows memory storage saving and the use of less computing capacity for the clustering computation. The clustering algorithm then runs over the reduced graph without considering the battery level anymore since it has already been taken into account in the reduction step ( $g(u) = 1$ ).

Fig. 4 shows the evolution of a clustering structure over **RNG**. We can see that first round elect nodes  $b$ ,  $d$  and  $f$  as *cluster-heads* (Fig. 4(a)). After a while, *cluster-heads* nodes have consumed more energy than the others leading to a new construction of the graph. On Fig. 4(b) we can see that links between  $a$  and  $b$  and between  $b$  and  $n$  for examples will now be avoided.

## VI. RESULTS

To evaluate the performances of the different versions of **BLAC**, we perform some simulations under the WSNET simulator<sup>2</sup>. We compare the four variants of **BLAC** to three close approaches of the literature: DDR [11] and density-based clustering [10] because of their algorithmic proximity to **BLAC** and LEACH [7] because of its energy efficiency concern. In order to observe different behaviors for LEACH, three values of the  $p$  parameter are used 5, 10 and 20% ( $p$  is the average number of *cluster-heads* in the network).

Nodes are placed using realistic maps from districts of Lille in north of France. Nodes are placed on streets and next to

<sup>2</sup><http://wsnet.gforge.inria.fr/>

building areas by using OpenStreetMap data<sup>3</sup> in order to get realistic topologies. Fig. 5 shows topologies we used to run the simulations, red dots represent nodes.

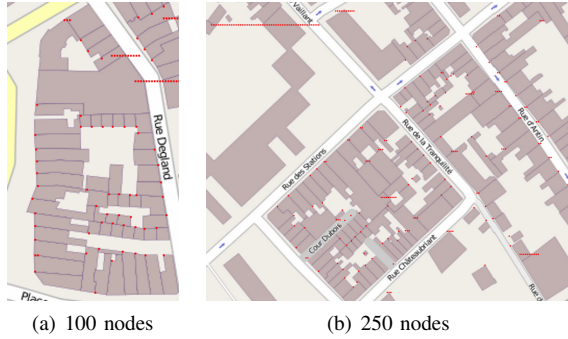


Fig. 5. Network topologies, each red dot represent a node.

In order to use a realistic model for transmitting and receiving costs in both peer-to-peer and GPRS communications, we consider the *Texas Instruments CC2420 ZigBee®* chip<sup>4</sup> and the *LEON-G100 GSM/GPRS* module from *u-blox*<sup>5</sup> respectively. The former one consumes 0.77 mW when idle, 35.46 mW for receiving (Rx), and 31.32 mW for transmitting (Tx). The latter one consumes 6.4 mW when idle and 1.25 W for receiving (Rx) or transmitting (Tx). A data traffic is also simulated. Each node generates 16 kbit of data periodically (every 5 seconds) and sends them to its parent. When a node receives data from a child, it stores them until it needs to send its own data and then sends the aggregated data to its own parent. For instance on Fig. 1(b), after node *a* has generated its own 16 kbit of data, it sends them to *b* which will store them until it generates in its turns 16 kbit of data. Then *b* sends 32 kbit to *m* which waits for its own data to send the aggregated 48 kbit to a base station using the GPRS radio chip.

At bootstrap, each node earns the same energy level set to 32 mWh of energy. When a node sends or receives a packet, the correct amount of energy is removed from the battery depending of the size of the message and data-rate of the chipset used.

#### A. Network lifetime

Fig. 6 illustrates the benefits of **BLAC** by showing the number of nodes alive regarding time.

We can notice that in solutions that do not consider battery level such as density-based and DDR, cluster-heads die quickly. When cluster-heads are dead, other nodes take their role till dying as well, and so on till there is no remaining alive node. This is why we observe an irregular decrease (kind of steps) in the number of alive nodes. Several nodes (the *cluster-heads*) die almost at the same time depending on the number of nodes they have in their cluster. LEACH performs much better, improving the lifetime up to 3 time longer compared to DDR and Density-based algorithms. Nevertheless, its performances greatly depend on the number of *cluster-heads* that have been set up. The more *cluster-heads*, the more nodes forwarding data (either as

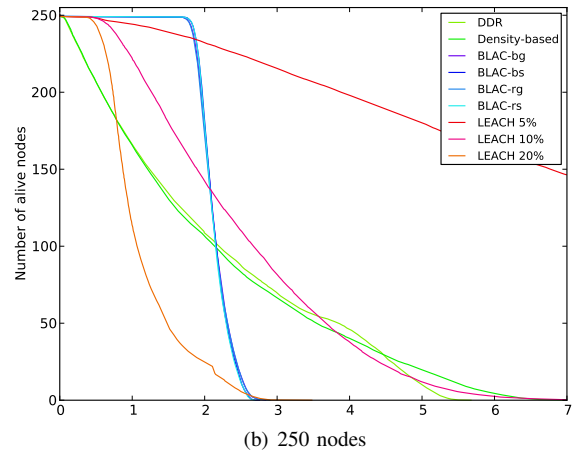
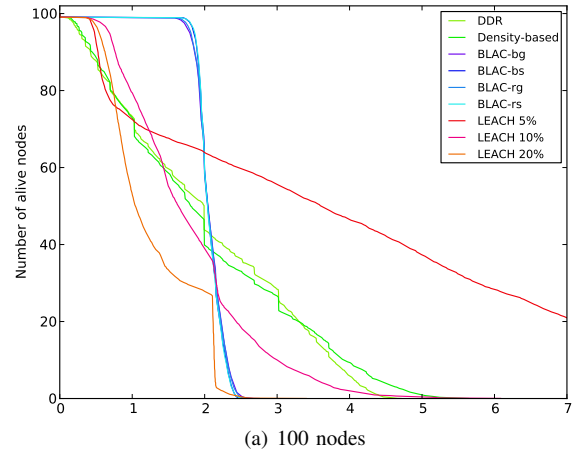


Fig. 6. Network lifetime

*cluster-head* or to *cluster-head*) and thus the shorter lifetime. **BLAC-\*** maintains the maximum number of alive nodes up to 900% longer than DDR and Density-based and up to 300% longer than LEACH as expected (see Section III). **BLAC-bg** and **BLAC-rg** present a slightly improvement compared to **BLAC-bs** and **BLAC-rs** resp. regarding the lifetime as they keep more nodes alive during the decreasing phase. This is due to the overall number of clusters built by every scheme. Indeed, **BLAC-bg** and **BLAC-rg** build less clusters than their corresponding variants and thus propose a different balance between the number and the size of *cluster-heads* that perform a little bit better.

In addition, note that the node synchronization cost has not been considered in the cost evaluation of LEACH.

#### B. Delivery ratio

Fig. 7 displays the delivery ratio of every algorithm with regards to time for 100 and 250 nodes. Delivery ratio is computed as the amount of data received divided by the amount of data sent. We observe that the different variants of LEACH loses slightly more data than **BLAC** algorithms regardless of the number of *cluster-heads*. Indeed, in LEACH, nodes without *cluster-head* can not send data so their data are lost. Even when the number of *cluster-heads* increases, some data are still lost because nodes need to share the medium with other nodes in

<sup>3</sup><http://www.openstreetmap.org>

<sup>4</sup><http://focus.ti.com/lit/ds/symlink/cc2420.pdf>

<sup>5</sup><http://www.u-blox.com/en/wireless-modules/gsm-gprs-modules/leon-gsm-module-family.html>

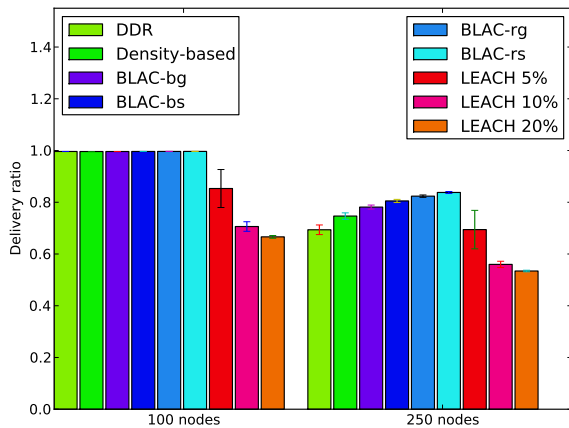


Fig. 7. Data delivery ratio

the same cluster and thus may not have enough time to transmit all data. In addition, since there are more *cluster-heads*, nodes die more quickly (see Fig. 6) since sending more data.

When the number of nodes increases in the network, every scheme loses data. For **BLAC** variants, density-based and DDR, this is due to the fact that when some intermediary nodes dies, the data it was holding for other are also lost. For LEACH, this is due to the fact that there are more and more nodes in a cluster that need to share the medium and thus, some of the data can not be sent on time.

### Discussion

We have seen through the results analysis that the four versions of **BLAC** are close even if **BLAC-rs** seems to offer the best performances in terms of quality of service (delivery ration VS network lifetime). Nevertheless, there are still performances that we can not evaluate here because of page restriction. The choice of the **BLAC** version must be made regarding the needs of the application concerning cluster size, stability and quality of service. For instance, if nodes are mobile **BLAC-bs** is the best choice because it offers a better stability against mobility as proven in [10]. If the size of clusters is not the main issue of the application, like in case of static nodes, **BLAC-rg** is the best choice for data delivery ratio.

## VII. CONCLUSIONS

In this paper, we have introduced a new family of clustering techniques. Two variants combine the battery level with the degree or the density as a metric for cluster creation. The two others apply a battery-level-based RNG construction. By completely integrating the battery level in the metric used to elect *cluster-heads*, **BLAC** balances energy consumption over nodes and maximizes the network lifetime. The algorithm is distributed and modifications due to network dynamics are handled locally, allowing scalability. Results show that our proposition improves network lifetime with no dead nodes up to 300% that is useful for applications like ours. For future work we will extend the comparison of **BLAC** with other algorithms like WCA [3]. Other energy models should also be used.

We will also run experimentations in real world by using for instance SensLAB<sup>6</sup> or Wisebed<sup>7</sup> platforms. Another field of investigation is range adjustment.

## REFERENCES

- [1] D. Baker and A. Ephremides. The architectural organization of a mobile radio network via a distributed algorithm. *IEEE Transactions on Communications*, 29(11), 1981.
- [2] O. Buyanjargal and Y. Kwon. AECC-Adaptive and Energy Efficient Clustering Algorithm for Content Based Wireless Sensor Networks. In *Inter. Conf. Computer Science and its Applications*, 2009.
- [3] M. Chatterjee, S. K. Das, and D. Turgut. WCA: A weighted clustering algorithm for mobile ad hoc networks. *Cluster Computing*, 5, 2002.
- [4] W. Dali and H.A. Chan. Clustering algorithm to balance and to reduce power consumptions for homogeneous sensor networks. In *Wireless Com., Networking and Mobile Computing*, 2007.
- [5] N. Dimokas, D. Katsaros, and Y. Manolopoulos. Energy-efficient distributed clustering in wireless sensor networks. *Journal of Parallel and Distributed Computing*, 70(4), 2010.
- [6] M. Gerla and J. Tzu-Chieh Tsai. Multiclustet, mobile, multimedia radio network. *Wireless Networks*, 1, 1995.
- [7] W.R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Hawaii Inter. Conf. on System Sciences*, 2000.
- [8] T. Kaur and Jinsuk B. A strategic deployment and cluster-header selection for wireless sensor networks. *IEEE Trans. on Consumer Electronics*, 55(4), 2009.
- [9] X. Min, S. Weiren, J. Chang-jiang, and Z. Ying. Energy efficient clustering algorithm for maximizing lifetime of wireless sensor networks. *Inter. Journal of Electronics and Com.*, 64(4), 2010.
- [10] N. Mitton, B. Sericola, S Tixeuil, E. Fleury, and I. Guérin Lassou. Self-stabilization in Self-organized Wireless Multihop Networks. *Ad Hoc and Sensor Wireless Networks*, 2011.
- [11] N. Nikaein, H. Labiod, and C. Bonnet. DDR-distributed dynamic routing algorithm for mobile ad hoc networks. In *Workshop on Mobile and Ad Hoc Networking and Computing*, 2000.
- [12] J. Radak, N. Mitton, and D. Simplot-Ryl. Using battery level as metric for graph planarization. In *AdHocNow*, Germany, 2011.
- [13] T. Shu, M. Krunz, and S. Vruthula. Power balanced coverage-time optimization for clustered wireless sensor networks. In *ACM inter. symp. on Mobile ad hoc networking and computing*, 2005.
- [14] G. T. Toussaint. The relative neighbourhood graph of a finite planar set. *Pattern Recognition*, 12, 1980.
- [15] O. Younis, M. Krunz, and S. Ramasubramanian. Node clustering in wireless sensor networks: recent developments and deployment challenges. *IEEE Network*, 20(3), 2006.

<sup>6</sup>www.senslab.info

<sup>7</sup>www.wisebed.eu