Evaluation Study of Self-Stabilizing Cluster-Head Election Criteria in WSNs

Mandicou Ba, Olivier Flauzac, Rafik Makhloufi and Florent Nolot
Université de Reims Champagne-Ardenne, France
CReSTIC - SysCom EA 3804
{mandicou.ba, olivier.flauzac, rafik.makhloufi, florent.nolot}@univ-reims.fr

Ibrahima Niang
Université Cheikh Anta Diop, Sénégal
Laboratoire d'Informatique de Dakar (LID)
iniang@ucad.sn

***Abstract*—In the context of Wireless Sensor Networks (WSNs), where sensors have limited energy power, it is necessary to carefully manage this scarce resource by saving communications. Clustering is considered as an effective scheme in increasing the scalability and lifetime of wireless sensor networks. We propose an energy-aware distributed self-stabilizing clustering protocol based on message-passing for heterogeneous wireless sensor networks. This protocol optimizes energy consumption and prolongs the network lifetime by minimizing the number of messages involved in the construction of clusters and by minimizing stabilization time. Our generic clustering protocol can be easily used for constructing clusters according to multiple criteria in the election of cluster-heads, such as nodes' identity, residual energy or degree. We propose to validate our approach under the different election metrics by evaluating its communication cost in terms of messages, stabilization time, energy consumption and number of clusters. Simulation results show that in terms of number of messages and energy consumption, it is better to use the Highest-ID metric for electing CHs. However, the criterion of energy provides a better distribution of clusters.**

***Keywords-Self-stabilizing clustering*; *Wireless Sensor Networks*; *Energy-aware*; *OMNeT++ simulator*.**

I. INTRODUCTION

Due to their properties and to their wide applications, Wireless Sensor Networks (WSNs) have been gaining growing interest in the last decades. These networks are used in various domains like: medical, scientific, environmental, military, security, agricultural, smart homes, etc. [1].

In a WSN, sensors have very limited energy resources due to their small size. This battery power is consumed by three operations: data sensing, communication, and processing. Communication by messages is the activity which needs the most important quantity of energy, while power required by CPU is minimal. For example, Pottie and Kaiser [2] shows that the energy cost of transmitting a 1KB message over a distance of 100 meters is approximately equivalent to the execution of 3 million CPU instructions by a 100 MIPS/W processor. Thus, saving communication power is more urgent in WSNs than optimizing processing. Consequently, to extend the sensor network lifetime, it is very important to carefully manage the very scarce battery power of sensors by limiting communications. This can be done through notably efficient routing protocols that optimize energy consumption. Many previous studies (e.g., Yu et *al.* [3] and Younis and Fahmy [4]) proved that clustering is an effective scheme in increasing the scalability and lifetime of wireless sensor networks. Clustering consists in partitioning the network into groups called clusters, thus giving a hierarchical structure [5].

Several clustering approaches are proposed in the literature and used, for example, in the case of a WSN for routing collected information to a base station. However, most of them are based on state model, so they are not realistic compared to message-passing based clustering ones. Moreover, approaches in the last category are not self-stabilizing and they are generally highly costly in terms of messages, while in the case of WSNs clustering aims at optimizing communications and energy consumption.

In this paper, we propose an energy-aware distributed self-stabilizing clustering protocol based on message-passing for heterogeneous wireless sensor networks. This protocol optimizes energy consumption and then prolongs the network lifetime by minimizing the number of messages involved in the construction of clusters and by minimizing stabilization time. It also offers an optimized structure for routing. Our clustering protocol is generic and complete. It can be easily used for constructing clusters according to multiple criteria in the election of cluster-heads such as: nodes' identity, residual energy, degree or a combination of these criteria. We propose to validate our approach by evaluating its communication cost in terms of messages, stabilization time, energy consumption and number of clusters. Thus, we compare its performance in the case of using different cluster-heads election methods under the same clustering approach and testing framework.

The remainder of the paper is organized as follows. Section II illustrates the related work on clustering approaches. Section III describes the proposed clustering approach, cluster-head election methods and the models used for representing both energy consumption and network structure. Section IV presents the validation of the proposed approach through simulation. Finally, Section V concludes this paper and presents our working perspectives.

II. RELATED WORK

Several self-stabilizing k-hops algorithms have been done in the literature [6], [7], [8].

Mitton et *al.* [6] applied self-stabilization principles over a clusterization protocol they proposed in [9] and presents properties of robustness. Each node calculates its density and

 ISBN: 978-1-61208-263-9

broadcasts it to its neighbors located at k-hops. This robustness is an issue related to the dynamicity of ad hoc networks, to reduce the time stabilization and to improve network stability.

Datta et *al.* [7], by using the criterion of minimal identity, have proposed a self-stabilizing distributed algorithm designed for the state model that computes a subset D is a minimal *k-dominating* set of graph G. By using D as the set of *clusterheads*, G is partitioned into clusters, each of radius k. This algorithm converges in $O(n)$ rounds and $O(n^2)$ steps and requires $\log(n)$ memory space per process, where n is the size of the network.

Caron et *al.* [8], by using an arbitrary metric, have proposed a self-stabilizing *k-clustering* algorithm based on a state model. Note that *k-clustering* of a graph is a partition of nodes into disjoint clusters, in which every node is at a distance of at most k from the clusterhead. This algorithm executes in $O(nk)$ rounds and requires $O(log(n) + log(k))$ memory space per process, where n is the network size.

These approaches are based in state model [7], [8] and are not realistic in the context of sensor networks. Furthermore, they have extremely high stabilization time.

The approach proposed by Mitton et *al.* [6], [9] generates a lot messages. The main reason is due to the fact that each node must know $\{k+1\}$-*Neighboring*, computes its *k-density* value and locally broadcasts it to all its *k-neighbors*. This is very expensive in terms of exchanged messages.

III. Proposed Clustering approach

A. Basic idea

To simplify the description of our approach, we consider the case where the selection criterion to become clusterhead is the node's identity. We will present later the proposed approach when using other CH election criteria.

Our proposed algorithm is self-stabilizing and does not require any initialization. Starting from any arbitrary configuration, with only one type of message exchanged, the nodes are structured in non-overlapping clusters in a finite number of steps. This message is called *hello message* and it is periodically exchanged between each neighbor nodes. It contains the following four information: node identity, cluster identity, node status and the distance to cluster-head. Note that cluster identity is also the identity of the cluster-head. Thus, the hello message structure is $hello(id_u, cl_u, status_u, dist_{(u,CH_u)})$. Furthermore, each node maintains a neighbor table $StateNeigh_u$ that contains the set of its neighboring nodes states. Whence, $StateNeigh_u[v]$ contains the states of nodes v neighbor of u.

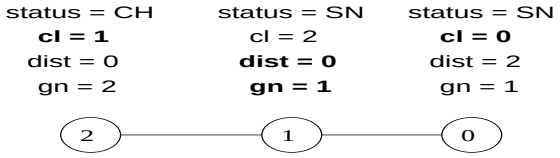
The solution that we propose proceeds as follows:

As soon as a node u receives a hello message, it executes three steps consecutively (see Algorithm 1). The first step is to update neighborhood, the next step is to manage the coherence and the last step is to build the clusters. During the last step each node u chosen as cluster-head the node which optimizes the criterion and located at most a distance k. After this three steps, u sends a hello message to its neighbors. The details of Algorithm 1 and mathematical proof are describe in Ba et *al.* [10].

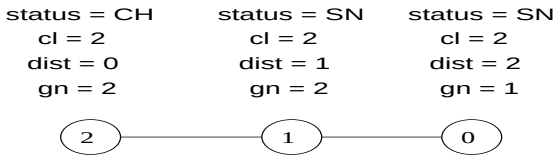
Algorithm 1: k-hops clustering algorithm

```
 /* Upon receiving message from a
    neighbor                                  */
1 UpdateNeighborhood();
2 CoherenceManagement();
3 Clustering();
```

After updating the neighborhood, nodes check their coherency. For example, as a cluster-head has the highest identity, if a node u has CH status, its cluster identity must be equal to its identity. In Fig. 1(a), node 2 is cluster-head. Its identity is 2 and its cluster identity is 1, so node 2 is not a coherent node. Similarly for nodes 1 and 0. Each node detects its incoherence and corrects it during the coherence management step. Fig. 1(b) shows nodes that are coherent.



(a) Incherence nodes



(b) Coherence nodes

Figure 1. Coherent and incoherent nodes

B. Cluster-heads election

Existing clustering approaches use one or more criteria for electing cluster-heads, for example: nodes' ID, degree, density, mobility, distance between nodes, service time as a CH, security, information features or a combination of multiple criteria. However, to the best of our knowledge, there is no paper in the literature where the same proposed approach is compared in the case of different CH election methods. It is important to study the influence of each criterion under the same test conditions and, ideally, under the same clustering approach. To this end, we propose a generic distributed self-stabilizing clustering approach that can be used with any CH election criterion and then we compare its cost and performance when considering important election criteria in the case of a WSN, namely: Highest-ID, Highest-degree and residual energy of nodes.

1) Highest ID:

Lowest-Identifier based clustering is originally proposed by Baker et *al.* [11]. It has proven one of the most performant clustering approaches in ad hoc networks [12], [13], [14], [15].

In our approach, each node compares its identity with those of its neighbors a distance 1. A node u elects itself as a cluster-head if it has the highest identity among all nodes of its cluster

(in Fig. 2, example of node 9 in cluster V_9). If a node u discovers a neighbor v with a highest identity then it becomes a node of the same cluster as v with SN status (in Fig. 2, example of nodes $1, 3, 4$ and 7 in cluster V_9). If u receives again a hello message from another neighbor which is into another cluster than v, the node u becomes gateway node with GN status (in Fig. 2, example of nodes 5 and 8 in cluster V_{10} and node 2 in cluster V_9). As the hello message contains the distance between each node u and its clusterhead, u knows if the diameter of cluster is reached. So it can choose another cluster as illustrated in Fig. 2.

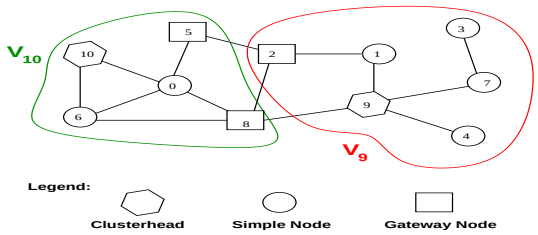


Figure 2. Clusters organization

2) Highest or Ideal Degree: In this approach, we determine how well suited a node is for becoming CH according to its degree D (i.e., the number of neighbors). There are two categories of approaches based on nodes' degrees. Some of them propose to limit communications by electing the node having the highest degree as CH. This is an original proposal of Gerla and Tsai [16]. However, each CH can ideally support only ρ (a pre-defined threshold) nodes to ensure an efficient functioning regarding delay and energy consumption. Indeed, at each step of the routing process, when a node has many neighbors it receives as many messages as its degree. This leads to a rapid draining of sensors' battery power. To ensure that a CH handles upto a certain number of nodes in its cluster, some approaches [14], [17], [18] propose to elect as CH the node having the nearest degree to an ideal value ρ. Thus, the best candidate is the one minimizing its distance to this ideal degree $\triangle_d = |D - \rho|$.

For the two cases described above, when more than one node has the maximum (respectively ideal) degree and is candidate to become a CH, the election is done according to a secondary criterion which is the highest ID. As each node of the network has a unique ID, this criterion is discriminating.

3) Residual Energy: In this approach, decision-making concerning the most suitable node to become CH is done according to the residual energy (i.e., remaining battery power level) of each sensor. Indeed, CHs are generally much more solicited during the routing process. So, in order to preserve their energy and to avoid the frequently reconstruction of the clusters, CHs need more important battery levels compared to the others normal nodes.

During the clustering procedure, network nodes progressively consume their energy due to the messages exchanges. Thus, after some rounds a node i with initially the maximum battery power level and candidate to become a CH can have later less energy than an another neighbor node j. This can lead to more iterations aiming at electing the other node j with the maximum residual energy. In order to limit the frequently changes of CH candidates for a negligible energy difference, we propose to use an energy gain threshold E_T. Thus, while $\triangle_e = |E_i - E_j|$ is less than E_T, the node i preserves its leadership position. This guarantees more stability of the clustering process and extends the network lifetime by minimizing the energy consumption involved in the clustering procedure.

C. Models

In order to implement our clustering approach in a realistic way, we use standard models for representing both the energy consumption and the network structure.

1) Energetic model: To model the energy consumption for a node when it sends/receives a message, we use the first order radio model proposed by Heinzelman et *al.* [19] and used in many other studies [3], [20], [21]. A sensor node consumes E_{Tx} amount of energy to transmit one l-bits message over a distance d (in meters). As shown in equation 1, when the distance is higher than a certain threshold d_0, a node consumes more energy according to a different energetic consumption model.

$$E_{Tx}(l,d) = \begin{cases} l * E_{elec} + l * \varepsilon_{fs} * d^2, & \text{if d} < d_0; \\ l * E_{elec} + l * \varepsilon_{mp} * d^4, & \text{if d} \geq d_0. \end{cases} \quad (1)$$

Each sensor node will consume E_{Rx} amount energy when receiving a message, as shown in equation 2.

$$E_{Rx}(l) = l * E_{elec} \quad (2)$$

The values of the parameters used in equations 1 and 2 to model energy are summarized in Table I:

TABLE I
RADIO MODELING PARAMETERS

Parameter	definition	Value
E_{elec}	Energy dissipation rate to run radio	$50nJ/bit$
ε_{fs}	Free space model of transmitter amplifier	$10pJ/bit/m^2$
ε_{mp}	Multi-path model of transmitter amplifier	$0.0013pJ/bit/m^4$
d_0	Distance threshold	$\sqrt{\varepsilon_{fs}/\varepsilon_{mp}}$

2) Network model: We consider a network represented by an arbitrary random graph based on Erdos Renyi model [22] with probability $p = 0,1$ for all network sizes. Our system can be modeled by an undirected graph $G = (V, E)$. $V = n$ is the set of network nodes and E represents all existing connections between nodes. An edge exists if and only if the distance between two nodes is less or equal than a fixed radius $r \leq d_0$. This r represents the radio transmission range which depends on wireless channel characteristics including transmission power. Accordingly, the neighborhood of a node u is defined by the set of nodes that are inside a circle with center at u and radius r and it is denoted by $N_r(u) = N_u = \{\forall v \in V \setminus \{u\} \mid d_{(u,v)} \leq r\}$. The degree of a node u in G is the number of edges which are connected to u, and it is equal to $deg(u) = |N_r(u)|$.

 ISBN: 978-1-61208-263-9

TABLE II
THEORETICAL COMPARISON OF STABILIZING TIME AND MEMORY SPACE

	Stabilizing Time	Memory space per node	Neighbourhood
Our approach	$n+2$	$\log(2n+k+3)$	1 hop
Datta et al. [7]	$O(n)$, $O(n^2)$	$\log(n)$	k hops
Caron et al. [8]	$O(n*k)$	$O(log(n)+log(k))$	k+1 hops

IV. VALIDATION FRAMEWORK

In this section, we present the evaluation study that we carried out using *ONMeT++* [23] simulator to compare the performance of the previously described clustering approach when utilizing different CH election methods. For generating random graphs, we have used the SNAP [24] library. All simulations were carried out using *Grid'5000* [25] platform.

A. Theoretical validation

In [10], we have provided a formal proof of our clustering approach. Table II illustrates a comparison of stabilizing time and memory space between our proposal algorithm and other approach designed for the state model. We note that our stabilization time does not depend on the parameter k contrary to approach proposed by Caron et *al.* [8]. We have a unique phase to discover the neighborhood and build *k-hops* clusters and an unique stabilizing time contrary to approach describes in [7]. Furthermore, we consider a 1-hop neighborhood at opposed to Datta et *al.* [7] and Caron et *al.* [8].

B. Testbed

The parameters we used in our simulations are summarized in Table III. In all our simulations, a 99% confidence interval I_c is computed for each average value represented in the curves. These intervals are plotted as error bars and computed according to this equation: $I_c = [\overline{x} - t_\alpha \frac{\delta}{\sqrt{n}}; \overline{x} + t_\alpha \frac{\delta}{\sqrt{n}}]$, where n is the population length, $\overline{x}$ is the average value, δ is the standard deviation, and finally, t_α has a fixed value of 2.58 in the case of 99% interval.

TABLE III
SIMULATION PARAMETERS

Parameter	Value
Message size	2000 bits
distance between 2 nodes	100 meters
Ideal degree	{5,20,50}
Energy threshold	{0.1%,0.01%}
Number of nodes	[100,1000]
Random graph model	Erdos Renyi
Network density	0.1
Number of simulations for each network size	100

C. Simulation results

1) Communication cost (messages): In order to evaluate the validity of our clustering approach, we first measure the necessary cost in terms of messages to achieve the clustering procedure.

Based on the same network topology, the clustering based on the criterion of ID generates less messages as shown in Figs 3 and 4. The main reason is that the ID criterion brings greater stability during the clustering phase. In addition, the ID criterion is simpler and deterministic compared to the criteria of degree or energy. Indeed, for the criterion of degree, it is necessary for nodes to receive a message from their neighbors to calculate their degree. Then, the degree is broadcasted and the clustering phase begins. This is expensive in terms of messages. Also, the criterion of energy ration generates more messages than the criteria of ID and degree. As energy is a parameter which decreases during the clustering phase, it provides less stability and requires more messages to reach a stable state in the entire network.

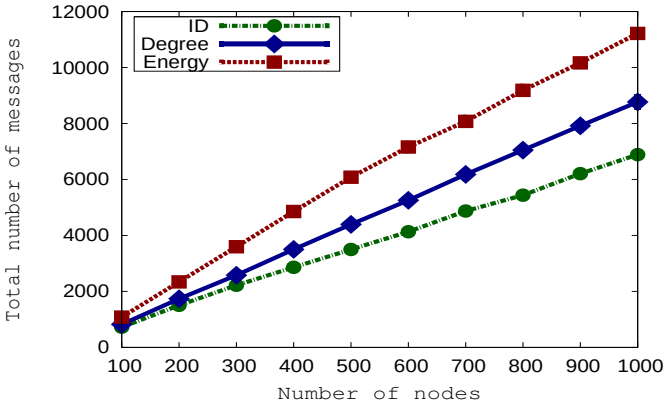


Figure 3. Total number of messages

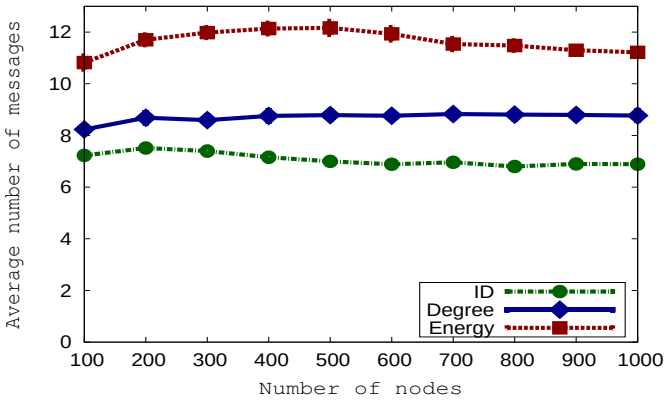


Figure 4. Average number of messages

2) Energy consumption: We have also measured the energy consumption required for building clusters in the entire

network. As illustrated in Figs. 5 and 6, the ID criterion consumes less energy during the clustering phase. Indeed, as illustrated in Figs. 3 and 4, both degree and energy generate more messages than ID during the construction of clusters. However, in sensor networks communications are the major source of energy consumption.

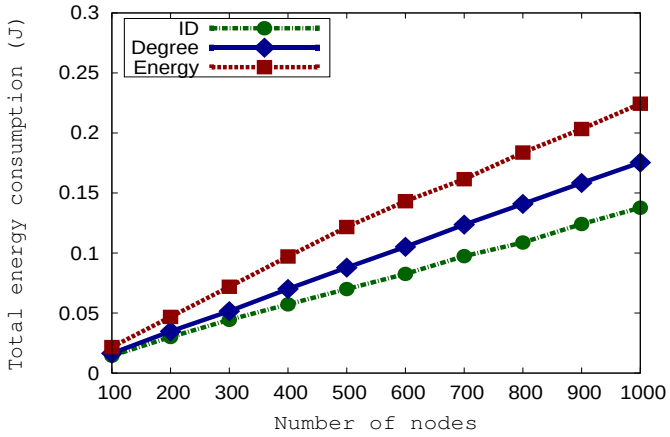


Figure 5. Total energy consumption

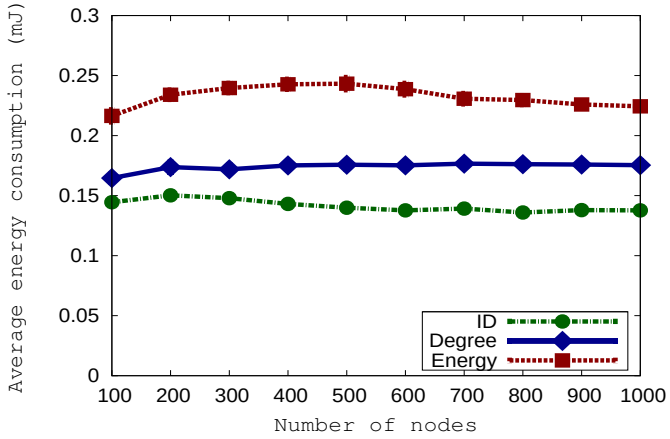


Figure 6. Average energy consumption

3) Number of clusters : The evaluation of the number of clusters as illustrated in Fig. 7 shows that the criterion of energy, even if it generates more messages and greater energy consumption, provides a better distribution of clusters in the network. The main reason is that the criterion of energy does not depend on the network topology contrary to for example the criterion of degree. In fact, in the latter, the node having the highest degree constructs large clusters.

4) Impact of highest and Ideal degree: To evaluate the impact of highest and Ideal degree, we fix $\triangle_d$ to 5, 20 and 40 and then we evaluate energy consumption and clusters distribution. We observe a slight increase in the energy consumption for ideal degree 5, 20 and 40 as illustrated in Fig. 8. Nevertheless, as illustrated in Fig. 9, the ideal degree offers a better distribution of the clusters. Note that the main problem

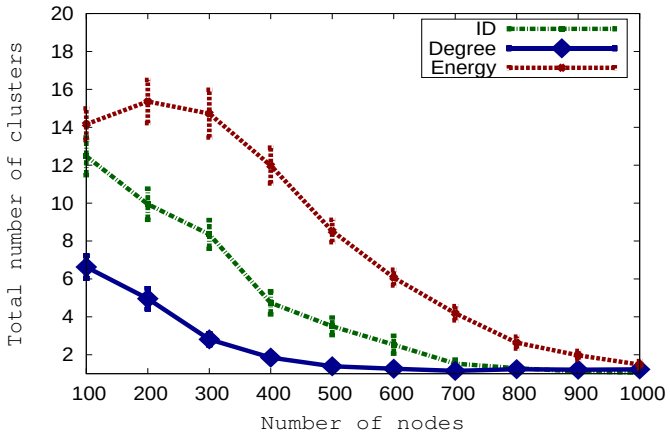


Figure 7. Number of clusters according to network size

with the highest degree is the distribution of clusters.

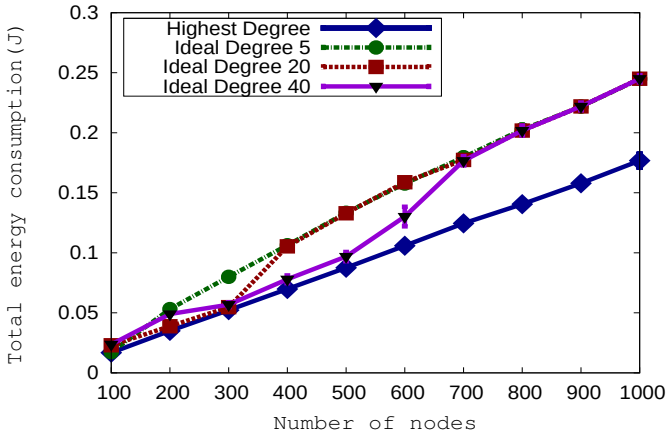


Figure 8. Energy consumption under highest and ideal degrees

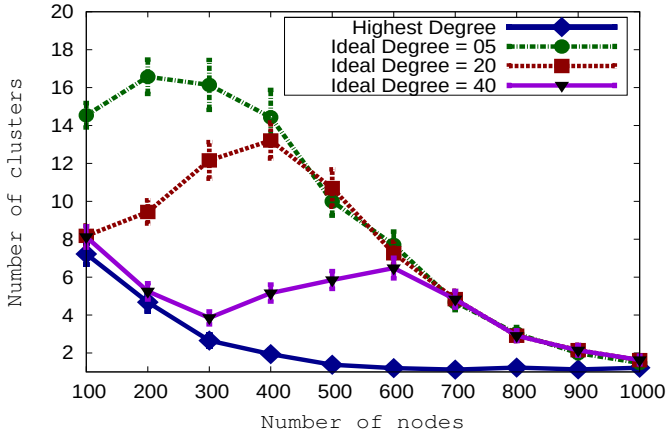


Figure 9. Number of clusters under highest and ideal degrees

5) Impact of residual energy or energy threshold: As the main problem with the criterion of energy is its volatility,

we fix energy threshold to limit abrupt changes of nodes when their energy CHs decreases substantially. We fixed the energy threshold to 0.1% and 0.01% and we evaluate both energy consumption and clusters distribution. Fig. 10 shows that energy threshold reduces energy consumption during the clustering phase. Indeed, the nodes no longer change after slight decrease of their energy CHs. This entails less messages exchanged and less energy consumption. Moreover, energy threshold offers a more balanced distribution of the clusters, as shows in Fig. 11.

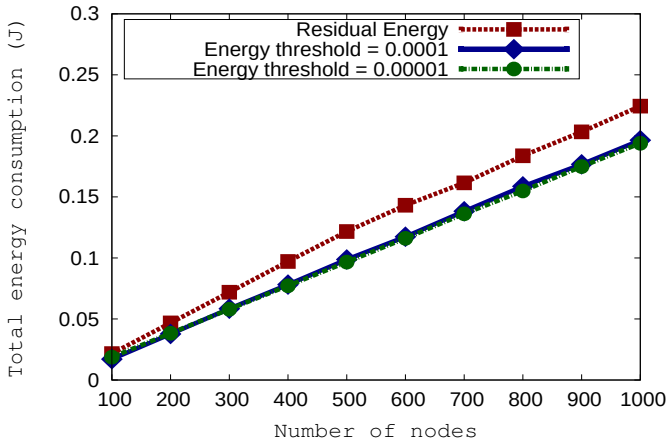


Figure 10. Residual energy *vs* Energy threshold

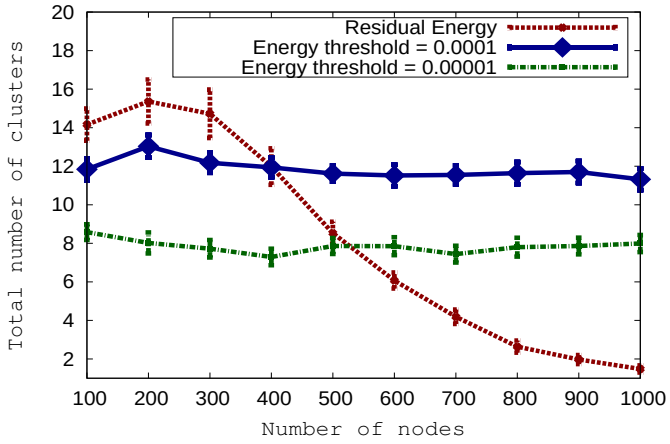


Figure 11. Number of clusters according to energy variation

V. Conclusion and Perspectives

In this paper, we proposed an efficient self-stabilizing distributed energy-aware clustering protocol for heterogeneous wireless sensor networks. This protocol prolongs the network lifetime by minimizing the energy consumption involved in the exchanged of messages. It can be used under different CHs election methods like those investigated in this work.

Simulation results show that in terms of number of messages and energy consumption, it is better to use the Highest-ID metric for electing CHs. However, the criterion of energy provides a better distribution of clusters.

As future work, we plan to propose routing process based on our clustering approach.

References

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," Computer Networks, pp. 393 – 422, 2002.

[2] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," Communications of the ACM, pp. 51–58, 2000.

[3] J. Yu, Y. Qi, G. Wang, and X. Gu, "A cluster-based routing protocol for wireless sensor networks with nonuniform node distribution," AEU - International Journal of Electronics and Communications, pp. 54 – 61, 2012.

[4] O. Younis and S. Fahmy, "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," IEEE Transactions on Mobile Computing, pp. 366–379, 2004.

[5] C. Johnen and L. Nguyen, "Self-stabilizing weight-based clustering algorithm for ad hoc sensor networks," in ALGOSENSORS, pp. 83–94, 2006.

[6] N. Mitton, E. Fleury, I. Guerin Lassous, and S. Tixeuil, "Self-stabilization in self-organized multihop wireless networks," in ICDCSW, pp. 909–915, 2005.

[7] A. K. Datta, S. Devismes, and L. L. Larmore, "A self-stabilizing O(n)-round k-clustering algorithm," in SRDS, 2009, pp. 147–155.

[8] E. Caron, A. K. Datta, B. Depardon, and L. L. Larmore, "A self-stabilizing k-clustering algorithm for weighted graphs," JPDC., pp. 1159–1173, 2010.

[9] N. Mitton, A. Busson, and E. Fleury, "Self-organization in large scale ad hoc networks," in MED-HOC-NET, 2004.

[10] M. Ba, O. Flauzac, B. S. Haggar, F. Nolot, and I. Niang, "Self-stabilizing k-hops clustering algorithm for wireless ad hoc networks," in 7th ACM IMCOM (ICUIMC), pp. 38:1–38:10, 2013.

[11] D. J. Baker and A. Ephremides, "The architectural organization of a mobile radio network via a distributed algorithm," IEEE Transactions on Communications, pp. 1694–1701, 1981.

[12] Y.-F. Wen, T. A. F. Anderson, and D. M. W. Powers, "On energy-efficient aggregation routing and scheduling in IEEE 802.15.4-based wireless sensor networks," WCMC, 2012.

[13] I. G. Shayeb, A. H. Hussein, and A. B. Nasoura, "A survey of clustering schemes for mobile ad-hoc network (MANET)," American Journal of Scientific Research, pp. 135–151, 2011.

[14] S. K. D. Mainak CHATTERJEE and D. TURGUT, "WCA: A weighted clustering algorithm (WCA) for mobile ad hoc networks," in Cluster Computing, pp. 193–204, 2002.

[15] C.-C. Chiang, M. Gerla, and L. Zhang, "Forwarding group multicast protocol (FGMP) for multihop, mobile wireless networks," Cluster Computing, pp. 187–196, 1998.

[16] M. Gerla and J. T.-C. Tsai, "Multicluster, mobile, multimedia radio network," Wireless Networks, pp. 255–265, 1995.

[17] W. Choi and M. Woo, "A distributed weighted clustering algorithm for mobile ad hoc networks," in AICT-ICIW, 2006.

[18] M. R. Brust, A. Andronache, and S. Rothkugel, "WACA: A hierarchical weighted clustering algorithm optimized for mobile hybrid networks," in ICWMC, 2007.

[19] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in HICSS, 2000.

[20] J. Wang, J.-U. Kim, L. Shu, Y. Niu, and S. Lee, "A distance-based energy aware routing algorithm for wireless sensor networks," Sensors, pp. 9493–9511, 2010.

[21] J. WANG, J. CHO, S. LEE, K.-C. CHEN, and Y.-K. LEE, "Hop-based energy aware routing algorithm for wireless sensor networks," IEICE Transactions on Communications, pp. 305–316, 2010.

[22] P. Erdos and A. Renyi, "On the evolution of random graphs," Publ. Math. Inst. Hung. Acad. Sci, pp. 17–61, 1960.

[23] A. Varga and R. Hornig, "An overview of the OMNeT++ simulation environment," in Simutools, pp. 60:1–60:10, 2008.

[24] SNAP: Stanford Network Analysis Platform. [Online]. Available: http://snap.stanford.edu

[25] F. Cappello and et *al.*, "Grid'5000: A large scale and highly reconfigurable grid experimental testbed," in GRID, pp. 99–106, 2005.

 ISBN: 978-1-61208-263-9