# A clustering method for wireless sensors networks

| S. Fouchal | Q. Monnet | D. Mansouri | L. Mokdad | M. Ioualalen |
|---|---|---|---|---|
| Laboratoire LSIIT (BFO) | Laboratoire LACL | Laboratoire LSI | Laboratoire LACL | Laboratoire LSI |
| University of Strasbourg | University of Paris-Est | USTHB, Algeria | University of Paris-Est | USTHB, Algeria |
| fouchal@unistra.fr | quentin.monnet@lacl.fr | mansouri.dj@gmail.com | lynda.mokdad@u-pec.fr | mioualalen@usthb.dz |

*Abstract*—[1] **Clustering algorithms have been widely used in many domains so as to partition a set of elements into several subsets, each subset (or "cluster") grouping elements which share some similarities. These algorithms are particularly useful in wireless sensor networks (WSNs), where they allow data aggregation and energy cuts. By forming clusters and electing cluster heads responsible for forwarding their packets, the small devices that compose WSNs have not to reach directly the base station (BS) of the network. They spare energy and they can lead further in time their measuring task, so as to detect forest fires or water pollution for example. In this paper, we will apply a new and general clustering algorithm, based on classificability and ultrametric properties, to a WSN. Our goal is to get clusters with a low computational complexity, but with an optimal structure regarding energy consumption.**

## I. INTRODUCTION

Clustering algorithms are often essential in wireless sensor networks to cut down on energy consumption and to improve scalability. Basically, they are algorithms which aim at dividing a set of element into a finite number of subsets. Those subsets are called "clusters", and in most cases they are formed in such a way that the similarity between the elements in a same subset is maximal, whereas the similarity between elements coming from different clusters is minimal. Partitioning (*i.e.* clustering) a set of element may serve various purposes. It can highlight differences between groups, and allows one to manipulate subclasses of items as single objects.

Clustering algorithms such as LEACH [1] or HEED [2] are particularly useful in building networks where energy or reliability of links are a concern. Wireless sensor networks (WSNs) are composed of a base station (BS) and of multiple devices. These devices are able to sense WSNs environment, to turn their measures into electric signals and to forward it by wireless transmission to the BS. Some regular measurement activities can not be led by human beings; in such cases, sensors are spread over wide areas to proceed to a constant watch. For instance, detecting forest fires, seismic activity, toxic gases or nuclear threats, evaluating water pollution, or detecting moving targets on a military field, are tasks that require the use of WSNs. Sensors must be able to resist to their hostile environment. They are designed to be small entities, which have a number of constraints: they are limited in computation ability as well as in available memory, their energy amount is finite, and they are not reloaded [3]. Once deployed, sensors must organize themselves to form a functional network that runs in autonomy.

Several problematics in WSNs have been widely studied in literature, such as network security [4], fault tolerance, or distributing the energy load [3]. To address these problems, existing solutions often include (but are not limited to) the use of a clustering algorithm, as for [5]. Indeed, the network is partitioned into several clusters, each containing a cluster head (CH), which is elected among the sensors and whose purpose is to forward the packets from its neighbor nodes to the BS. In terms of scalability, clustering makes easier adding or withdrawing new sets of sensors to/from the network. As for the fault tolerance or the communication security, it may be easier to scale a solution to a cluster rather than to the whole network. Regarding energy consumption, it is far less consuming if each "normal" node sends data to the "closest" CH rather than emitting directly to the BS. Thus CHs are the only nodes to carry out the energy-expensive communication with the BS, after having possibly proceeded to data fusion and/or aggregation [6] [7].

For these reasons a wide number of clustering algorithms have been proposed during the last decades [8] [9]. Some of them are general and may be applied in a variety of domains.

This is the case, for example, for the *fast and flexible unsupervised clustering algorithm based on ultrametric properties* (FFUCA) proposed by Fouchal and Lavallée in [10]. Ultrametric spaces are ordered spaces endowed with a "distance" application which verifies the strong triangle inequality. Basically, one could consider that items from a same cluster are "equidistant" to those of another one. From a random sample data, the proposed algorithm provides an ultrametric space, using a distance, to get an outline of the proximity between elements w.r.t to the used distance. Then it is expanded so as to partition the whole set of elements. The computational complexity of this algorithm in the (rare) worst case is $\mathcal{O}(n^2) + \mathcal{O}(m^2)$, where $m$ is the size of the sample data and $n$ the number of elements in the whole processed set. In most cases, the computational cost is $\mathcal{O}(n) + \mathcal{O}(m^2)$, or rather $\mathcal{O}(n) + \epsilon$ as $m$ is insignificant.

In this paper, we will apply the FFUCA to a WSN. We will define a judicious distance between sensors of the WSN, and ensure that it follows metric properties. The use of the FFUCA algorithm provides an accurate way to cluster the network according to desired properties with an optimal computational cost. In our case, physical distance between the sensors in

a cluster is an important parameter: as it comes longer, the radio signal to communicate between two nodes must be more powerful, and more energy is consumed. Eventually, our goal is to save energy and to enable the WSN to live longer. The paper is organized as follows. Section II, describes our approach of clustering in WSNs based on FFUCA. We summarize the main contribution of this study and we give indications on future developments of our future work in Section III.

## II. CONSIDERED METHOD

Building an organizational structure in Wireless Sensor Networks (WSNs) is a known problem. Since the fundamental advantage of network structure is the ability to deploy sensors in an ad hoc manner, it is difficult to organize nodes into groups pre-deployment [9].

The clustering process is an important part in organizing WSNs. In fact, it can affect dramatically the network performance. Whatever, there exists several specific approaches to cluster WSNs which should consider the following conditions [9]:

- *Limited energy & network lifetime:* sensor energy in the network is limited. The proper management of this resource is vital to the network lifetime.
- *Network ability:* the performance of the WSN is tied to an optimal organization of sensor nodes and cluster heads which improve their ability by decreasing their energy consumption.
- *Application dependency:* the clustering scheme must be flexible to the variety of application requirements.

Considering these requirements we propose a new approach of clustering in WSNs based on FFUCA (Fast and Flexible Unsupervised Clustering Method) method [10]. FFUCA is a general clustering method based on classificability and ultrametric properties. It has a computational cost of $\mathcal{O}(n) + \epsilon^2$ in the average case and $\mathcal{O}(n^2) + \epsilon$ in the (rare) worst case. Using FFUCA allows a fast and optimal (regarding energy consumption) organization of clusters, cluster heads and sensor nodes in the network. The resulting structure is based essentially on a energy consumption measure $d_e$ which ensures a proper economy of energy (see properties bellow). Thus, it increases the network lifetime and improves its performance. The FFUCA is flexible to the processed data (nodes, energy...) and to the user settings (size of sample nodes and values of critical thresholds of vicinity). Indeed, the user is an actor in the execution of the algorithm. He can parameter the entries and refine the results.

The general idea of FFCUA method is based on two principal steps:

1) deducing the behavior of the whole sensors with the energy consumption measure $d_e$ from sample nodes. The sample is chosen uniformly at random (whose size is given by the user). Inferring the behavior consists in

building an ultrametric (ordered) space[3] which is structured hierarchically (see [10] [18] for further details). The behavior includes seeds of clusters and dense groups of sensors (cluster sizes). The generated seeds are the best candidate nodes to be cluster heads (CHs);
2) aggregating the rest of sensors (except those chosen as sample) in clusters according to information of the step 1 (*i.e.* cluster seeds and thresholds).

### A. Principle

*Assumption 1:* Consider a given WSN $N$ endowed with an energy consumption measure $d_e$ which satisfies the following properties of distance (metric) for all any three sensors $s_1, s_2, s_3 \in N$ (see Figure 1):

1) (Symmetry) $d_e(s_1, s_2) = d_e(s_2, s_1)$.
   The required energy to transmit one bit of data from $s_1$ to $s_2$, is the same as to transmit one bit from $s_2$ to $s_1$;
2) (Positive definiteness) $d_e(s_1, s_2) \geqslant 0$, and $d_e(s_1, s_2) = 0 \Leftrightarrow s_1 = s_2$. The required energy for the transmission of one bit of data is $\geqslant 0$. The transmission of one bit from $s_1$ to $s_1$ is null;
3) (Triangle inequality) $d_e(s_1, s_2) \leqslant d_e(s_1, s_3) + d_e(s_3, s_2)$. The required energy to transmit one bit of data from $s_1$ to $s_2$ is $\leqslant$ to the sum of required energy to transmit one bit of data from $s_1$ to $s_3$ and from $s_3$ to $s_2$.
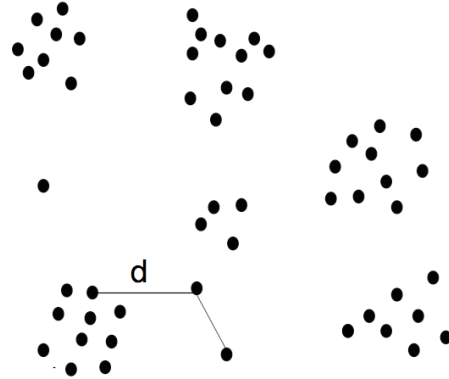


Figure 1. Example of metric space: a WSN described by a distance $d$

The FFUCA method is composed of the following steps:

*Step 1:* Choose uniformly at random a sample nodes from the network $N$ of size $m$ (*e.g.* $m = \frac{n}{50}$ nodes if $n = 1,000$) (see Figure 2)

*Remark 1:* The choice of the sample nodes depends on the processed network and on the expert in its deployment.

*Remark 2:* The number $m$ of nodes in the sample depends on the number $n$ of nodes in the whole WSN compared to the bounds of $d_e$. The larger $n$ is the pettier $m$ is compared to that of $n$ [19].

---

[2]Define the clustering in $\mathcal{O}(m^2)$ of sample of $m$ nodes, $m$ is petty compared to global number of nodes in the processed WSN.

[3]A set of nodes endowed with an ultrametric measure, it can be represented by an valued tree.
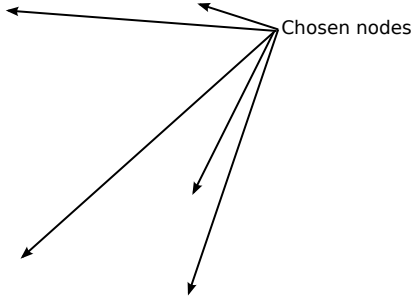
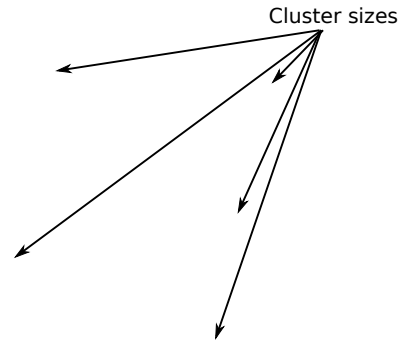Figure 2.  Chosen nodes (with $m = 5$)



Figure 4.  Choosing cluster sizes, represented by circles.

*Example 1:* Consider a WSN of size $n = 100,000$ nodes and distance $d_e \in [0, 0.5]$; if we use a sample of 50 nodes, we can easily get an idea about an optimal organization of nodes w.r.t energy consumption. But if $n = 500$ and $d_e \in [0, 300]$, choosing a sample of 5 nodes does not provide enough information about the manner the nodes behave with $d_e$.

*Step 2:* Execute a classic hierarchical clustering algorithm (*WPGMA* or *UPGMA*, two known algorithms for building ultrametric spaces) using the distance $d_e$ on the chosen sample nodes.

*Step 3:* Represent the distances in the resulting dendrogram, thus the ultrametric space is built (see Figure 3).
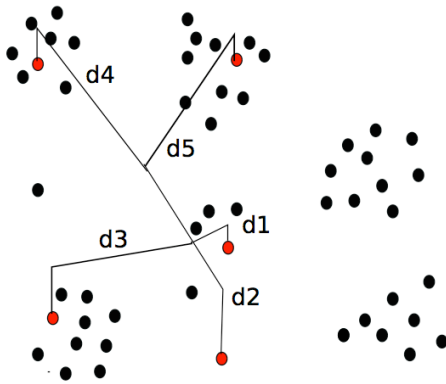


Figure 3.  Building an ultrametric space from the chosen sample nodes and representing distances.

*Step 4:* Deduce clusters' intervals (thresholds) which depend on the vicinity sought-after and $d_e$, as large or precise view of proximities between nodes (this step is specified by the user or expert) (see Figure 4).

*Step 5:* Choose uniformly at random one representative node per cluster from the result of the last step (see Figure 5).
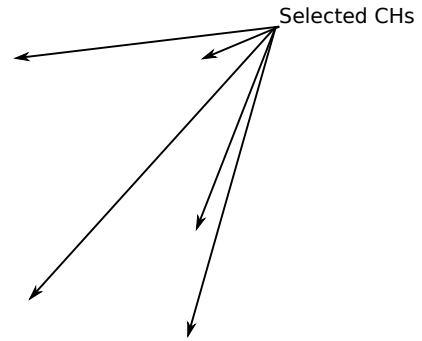


Figure 5.  Choosing CHs

*Remark 3:* In this example, as the number of nodes is small we use the same nodes chosen for sample as cluster representative nodes.

*Step 6:* Pick the rest of nodes one by one and compare them (according to $d_e$) with the cluster representative nodes of the previous step:

- If the compared data is close to one (or more) representative node(s) w.r.t. the defined thresholds, then add it to the same cluster(s) as this(those) representative node(s);
  *Remark 4:* If the compared data is close to more than one representative, then it will be added to more than one cluster and consequently generate an overlapping clustering (see [20] [21] for details about overlapping clustering).
  *Remark 5:* The random choice of initial points does not affect the resulting clusters [10] [18].
- Else, create a new cluster which will be represented by the remote node (see Figure 6).

*Remark 6:* Outliers are farthermost nodes, they are considered as noises in general. FFUCA is sensitive to this kind of
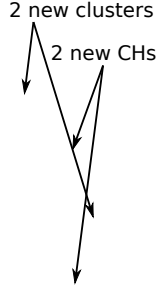
2 new clusters

2 new CHs

Figure 6.   Nodes aggregation and new clusters emergence
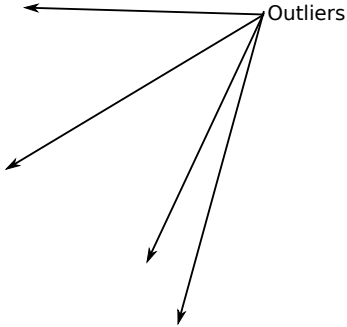
nodes [10] (see Figure 7).



Outliers

Figure 7.   Detection of outliers

The computational cost of FFUCA method is, in the average case, equal to $\mathcal{O}(n)$, and in the rare worst case, equal to $\mathcal{O}(n^2)$. We ignore $\epsilon$ (the cost of the sample nodes' hierarchical clustering) because it is unimportant compared to the global number of nodes in the processed network. The complexity of FFUCA remains the same even if the size of the network increases, thus it can process dynamic WSNs [10].

*B. Algorithm*

```
Procedure FFUCA
 • Variables:
      Metric space:
      a. Wireless Sensor Network N of n
         nodes;
      b. Energy consumption measure
         (metric) d_e.
 • Begin:
  1) Choose uniformly at random a part of
     nodes as sample;
  2) Build an ultrametric space from the
     sample nodes, using the hierarchical
     clustering algorithm WPGMA (or
     UPGMA);
     //complexity O(m²) = ϵ
     //where m = sample nodes size
  3) Determine intervals and cluster
     seeds (cluster representative nodes)
     from the hierarchy of step 2;
  4) For i < n − k; i++:
     //k = number of representative nodes
     – For j < k; j++:
          Calculate d_e(i,j);
          //Complexity = O(k × n), in rare
          //worst case it equals O(n²),
          //provinding only single nodes.
          * If d_e(i,j)     ⩽ interval of the
            cluster of j
          * Then: allocate i to the
            cluster of j;
            · If i ∈ more than one cluster
            · Then: keep it only in the
              cluster of the closest
              representative;
              //A node can belong to 2 (or
              //more) clusters if it is
              //equidistant to 2 (or more)
              //representative nodes.
            · End If
          * Else: Create a new cluster and
            allocate i to this cluster.
            //The new cluster will be
            //represented by the remote
            //node.
          * End If
     – End For
  5) End For
 • End
```

## III. Conclusion

We propose in this paper a novel approach of clustering in WSNs based on the FFUCA method and on a metric measure of energy consumption which provide rapidly an optimal (regarding energy consumption) organization of nodes. The complexity of our approach is linear ($\mathcal{O}(n)$ in average case and $\mathcal{O}(n^2)$ in the rare worst case), consequently it can process large networks. In addition, this computational cost remains the same even if the size of the network increases, thus it has the ability to cluster dynamic WSNs. Thanks to its simplicity, this approach can easily be applied. It can also be used as a hierarchical clustering method, thus providing a clearer organization of nodes.

## References

[1] M. J. Handy, M. Haase, and D. Timmerman. Low energy adaptive clustering hierarchy with deterministic cluster-head selection. In *Proceedings of the 4th IEEE Conference on Mobile and Wireless Communications Networks*, Stockholm, Sweden, 2002.

[2] O. Younis and S. Fahmy. HEED: a hybrid, energy-efficient distributed clustering approach for ad hoc sensor networks. *IEEE Trans. Mobile Comput.*, 3(4):366–379, 2004.

[3] G. Anastasi et al. Energy conservation in wireless sensor networks: A survey. *Ad Hocs Networks*, 7:537–668, 2009.

[4] F. Hu and N. Sharma. Security considerations in ad hoc sensor networks. *Ad Hoc Networks*, 3:69–89, 2005.

[5] M. Guechari, L. Mokdad, and S. Tan. Dynamic solution for detecting denial of service attacks in wireless sensor networks. In *IEEE International Conference on communications*, June 2012.

[6] S. Ozdemir and Y. Xiao. Secure data aggregation in wireless sensor networks: A comprehensive overview. *Computer Networks*, 53:2022–2037, 2009.

[7] Y. Liang. Efficient temporal compression in wireless sensor networks. In *Proceedings of the 36th Annual IEEE Conference on Local Computer Networks*, Bonn, Germany, 2011.

[8] A. A. Abbasi and M. Younis. A survey on clustering algorithms for wireless sensor networks. *Computer Communications*, 30:2826–2841, June 2007.

[9] D. J. Dechene, A. El Jardali, M. Luccini, and A. Sauer. A survey of clustering algorithms for wireless sensor networks. project report, December 2006.

[10] S. Fouchal and I. Lavallée. Fast and flexible unsupervised clustering algorithm based on ultrametric properties. In *Proceedings of the 7th ACM Symposium on QoS and security for Wireless and Mobile Networks*, Miami, USA, 2011.

[11] W. R. Heinzelman et al. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the IEEE Hawaii international conference on system sciences*, 2000.

[12] Lehsaini M. *Diffusion et couverture basées sur le clustering dans les réseaux de capteurs : application à la domotique*. Phd. thesis, University of Franche-Comté, 2009.

[13] M. Ye, C. Li, G. Chen, and J. Wu. Eecs: an energy efficient clustering scheme in wireless sensor networks. In *Proceedings of the 24th International Performance, Computing, and Communications Conference*, volume 1, pages 535–540. IEEE, 2005.

[14] F. Murtagh. A survey of recent advances in hierarchical clustering algorithms. *The Computer Journal*, 26(4), 1983.

[15] D. Baker and A. Ephremides. The architectural organization of a mobile radio network via a distributed algorithm. *IEEE Transactions on Communications*, 29(11):1694–1701, 1981.

[16] A. D. Amis, R. Prakash, T. H. P. Vuong, and D. T. Huynh. Max-min d-cluster formation in wireless ad hoc networks. In *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 1, pages 32–41. IEEE, 2000.

[17] G. Cleuziou. An extended version of the k-means method for overlapping clustering. In *Proceedings of the 9th International Conference on Pattern Recognition*, 2008.

[18] S. Fouchal, M Ahat, and I. Lavallée. Optimal clustering method in ultrametric spaces. In *Proceedings of the 16th IEEE Symposium on Computers and Communications*, pages 123–128, Kerkyra (Corfu), Greece, June 2011. IEEE Computer Society.

[19] K. Bayer, J. Goldstein, R. Ramakrishan, and U. Shaft. When is nearest neighbor meaningful? In Springer-Verlag, editor, *Proceedings of the 7th International Conference on Database Theory LNCS*, pages 217–235, Miami, USA, 1999.

[20] F. Brucker. *Modèles de classification en classes empiétantes*. Phd. thesis, Dep. IASC of École Supérieure des Télécommunications de Bretagne, 2001.

[21] G. Cleuziou. *Une méthode de classification non-supervisée pour l'apprentissage de règles et la recherche d'information*. Phd. thesis, University of Orléans, 2006.