

Un algorithme hybride pour le choix des cluster-head dans les réseaux de capteurs sans fil

Cheikh Sidy Mouhamed CISSE*, Cheikh SARR*, Hassan MAHAMAT BORGOU*,

*Université de Thiès, Unité de Formation et de Recherche
en Sciences et Technologies, Département Informatique

cheikhsidy.cisse@univ-thies.sn - csarr@univ-thies.sn, mahamatborgou@yahoo.fr

Résumé— Les réseaux de capteurs sans fil sont composés d'un grand nombre de nœuds qui sont des micro-capteurs intelligents capables de récupérer des informations sur leur environnement immédiat. La technique de clustering consiste à organiser le réseau en identifiant un sous-ensemble dominant de nœuds appelés cluster-head (CH). Ces derniers sont responsables de la collecte des informations auprès des autres nœuds et de leur envoi vers une station réceptrice appelée « station de base ». Concevoir et développer des algorithmes efficaces de choix de cluster-head dans les réseaux de capteurs sans fil hiérarchiques s'avère être une tâche cruciale pour améliorer les performances du réseau. Dans ce papier, nous proposons HCA (Hybrid Clustering Algorithm), un nouvel algorithme distribué de choix de cluster-head basé sur une métrique hybride prenant en compte trois paramètres : le voisinage, l'énergie consommée et la distance d'un capteur par rapport à la station de base (SB). Les résultats des simulations montrent que HCA permet d'augmenter la durée de vie du réseau et la quantité de données agrégées au niveau du réseau.

Mot clés— Réseaux de capteurs sans fil, cluster-head, algorithme distribué, énergie, voisinage, distance.

I. INTRODUCTION

Un réseau de capteurs sans fil (RCSF) est un réseau ad hoc constitué d'un grand nombre de micro-capteurs capables de récolter et de transmettre des données de façon autonome [1]. Chaque nœud est composé d'une unité de traitement pour l'acquisition des données, d'un émetteur-récepteur pour l'envoi et la réception des données collectées et d'une source d'énergie qui fournit l'énergie nécessaire au fonctionnement global du système (en générale une batterie) [3]. Les données sont transmises via plusieurs sauts vers un puits (également appelé point de collecte ou collecteur) [2] appelé aussi station de base.

Les RCSF ont des ressources énergétiques limitées et généralement des batteries non-rechargeables. L'économie d'énergie est par conséquent une question à prendre en compte dans la conception du réseau étant donné qu'elle influe directement sur la durée de vie des nœuds, voire du réseau tout entier. C'est dans le but d'organiser et d'améliorer les performances du réseau que le clustering est devenu une technique fondamentale. Le clustering peut être définie comme un processus de regroupement par sous-ensembles de nœuds dans un réseau afin de leur attribuer un chef (un cluster-head). Ce dernier a pour rôle d'agréger les données en provenance des

autres nœuds du cluster. Le cluster-head peut ensuite transmettre ces données à un puits grâce à un moyen de communication longue distance. La technique de clustering a largement été discutée par la communauté des chercheurs pour améliorer l'évolutivité du réseau [4]. Par conséquent, il apparaît clairement que les nœuds choisis comme cluster-head doivent posséder des ressources supérieures à celles des autres nœuds du cluster qu'ils gèrent.

L'objectif de ce papier est de proposer un algorithme distribué permettant de choisir ces nœuds cluster-head. Ce choix se base sur une métrique hybride prenant en compte trois paramètres : le voisinage à un saut, la quantité d'énergie consommée et la distance des nœuds par rapport à la station de base. L'élection des cluster-head se fait de manière totalement dynamique, un nœud simple pouvant devenir cluster-head et vice-versa en fonction de la valeur prise par la métrique hybride dans le temps.

Dans la section II, nous présentons un état de l'art sur des techniques de clustering dans les RCSF. Dans la section III, nous décrivons le protocole HCA et enfin, nous comparons les performances de notre algorithme avec d'autres approches existantes dans la section IV.

II. ETAT DE L'ART

Les chercheurs ont proposé différentes techniques pour la formation des *clusters* et la sélection des *cluster-heads*. Chacune de ces solutions permet d'identifier un sous-ensemble de nœuds dans le réseau pour lui attribuer un leader.

Heinzelman et al. [5] ont proposé un algorithme de clustering distribué appelé LEACH pour le routage dans les réseaux de capteurs sans fil hiérarchiques. L'élection des *cluster-heads* se fonde sur une génération d'un nombre aléatoire et attribue ce rôle aux différents nœuds selon la politique de gestion *Round-Robin* (c'est-à-dire tourniquet) afin de garantir une dissipation équitable d'énergie entre les nœuds. Les *rounds* ont approximativement le même intervalle de temps déterminé préalablement comme stipulé dans [6]. Dans le but de réduire la quantité d'informations transmises à la station de base, les *cluster-heads* agrègent les données capturées par les nœuds membres qui appartiennent à leur propre cluster, et envoient un paquet agrégé à la station de base. Si le CH meurt, tous les autres nœuds qui appartiennent au *cluster* ne pourront plus communiquer les informations qu'ils ont capturées durant le *round* courant. Chaque round est composé de deux phases. La première nommée *Set-Up-Phase* durant laquelle le réseau

s'auto-organise en *clusters* et les *clusters-heads* sont formés. La deuxième phase nommée *Study-State-Phase*, durant laquelle les données sont collectées et routées vers la station de base. Afin d'améliorer LEACH, d'autres propositions ont été faites dans la littérature telles que : TEEN [12], LEACH-C [13], APTEEN [14], PEGASIS [15], HEED [16], LEACH-G [7].

WCA (Weighted Clustering Algorithm) [8] est un algorithme de clustering basé sur la combinaison de quatre métriques pour le calcul du poids d'un nœud. Ces métriques sont : la différence de degré D_u , la somme des distances entre u et chacun de ses voisins P_u , la mobilité relative moyenne M_u et le temps qu'un nœud reste cluster-head Δ_u . Chacune de ces métriques est pondérée par un coefficient w_i et le poids d'un nœud est obtenu par la formule suivante :

$$W_u = w_1 \Delta_u + w_2 D_u + w_3 M_u + w_4 P_u$$

$$\text{avec } w_1 + w_2 + w_3 + w_4 = 1$$

L'algorithme WCA propose des clusters de taille maximale égale à M nœuds. Le nœud ayant le plus petit poids W_u est choisi comme *cluster-head*.

Les auteurs de [9] ont proposé un algorithme de clustering nommé NQCA (Node Quality based Clustered Algorithm) dont le but est d'améliorer les performances de l'algorithme WCA en s'appuyant sur la combinaison de deux modèles de clustering. Le premier modèle consiste à ne pas prendre comme cluster-head les nœuds singleton mais uniquement les nœuds qui ont au minimum trois voisins. Le deuxième modèle se base sur la distance des nœuds par rapport au CH pour maintenir la stabilité du cluster en divisant la plage de transmission du nœud CH en trois zones. La première zone contient les nœuds qui sont les plus proches du CH, la deuxième zone est une zone intermédiaire tandis que la dernière zone appelée « zone à risque » est considérée comme défavorable. Les deux premières zones appelées « zone de confiance » favorisent la stabilité du cluster. Dans le modèle précédent, le nœud CH garde son statut aussi longtemps que qu'il y a de nœuds dans la zone de confiance. La combinaison de ces deux modèles permet d'élire un nœud comme cluster-head et de le garder dans un cluster pour une certaine durée.

BLAC (Battery-Level Aware Clustering) [10], est une famille d'algorithmes de clustering qui utilise le niveau d'énergie associée à un autre paramètre comme le degré ou la densité comme métrique pour élire un nœud comme cluster-head. L'objectif est de prolonger la durée de vie du réseau. Le rôle de *cluster-head* est assuré par chaque nœud à tour de rôle afin d'équilibrer l'énergie consommée sur l'ensemble des nœuds sans pour autant nécessiter de grosses modifications ou des messages de contrôle supplémentaires.

L'algorithme « Hybrid Clustering Algorithm for Optimal Clusters in Wireless Sensor Networks » [11] a pour objectif de trouver un nombre optimal de clusters pour un ensemble de nœuds distribués sur une zone géographique afin de sélectionner un cluster-head au niveau de chaque cluster. Dans le but de minimiser la consommation d'énergie de chaque micro-capteur et afin d'accroître globalement la durée de vie du réseau, nous proposons l'algorithme HCA dont l'originalité repose sur une métrique hybride combinant le voisinage, la

distance des nœuds par rapport à la SB et l'énergie consommée par chaque nœud. L'algorithme HCA fournit également un équilibrage énergétique à travers les nœuds du réseau. En effet, les nœuds changent alors naturellement de cluster-head en fonction de la valeur de la métrique.

III. HCA : UN ALGORITHME DE CLUSTERING

Nous présentons dans cette section notre algorithme hybride de clustering. Chaque nœud appartenant à un cluster envoie ses données au cluster-head qui les agrège et les envoie directement à la station de base. Cependant, avant de décrire l'algorithme d'élection des CH, nous allons d'abord définir la métrique hybride sur laquelle repose HCA.

1. Métrique hybride

Pour élire un nœud u comme CH, nous utilisons une métrique hybride $m(u)$. Cette dernière est obtenue en se basant sur les paramètres suivants :

- $|N(u)|$, le nombre de nœud dans le voisinage de u
- $dist(u, b)$, la distance euclidienne de u par rapport à la SB b .
- $Er(u)$, énergie résiduelle du nœud u

Nous utilisons aussi les notations suivantes :

- N le nombre maximal de nœuds par cluster
- L_{\max} la distance maximale entre deux nœuds (qui représente le diagonal du carré).
- $E(u)$, l'énergie initiale du nœud u

Nous définissons la métrique $m(u)$ du nœud u par :

$$m(u) = \frac{|N(u)|}{N} + \frac{dist(u, b)}{L_{\max}} + \frac{E(u) - E_r(u)}{E(u)} \quad (1)$$

Pour une identification plus aisée, nous définissons trois nouvelles variables ainsi :

$$\Delta_1 = \frac{|N(u)|}{N}, \Delta_2 = \frac{dist(u, b)}{L_{\max}} \text{ et } \Delta_3 = \frac{E(u) - E_r(u)}{E(u)}$$

Le nœud u ayant la métrique la plus faible dans son voisinage, a la plus grande probabilité d'être élu comme CH. Par conséquent, diminuer la valeur de la métrique $m(u)$ revient à minimiser les trois variables Δ_1 , Δ_2 et Δ_3 .

Minimiser la valeur de Δ_1 revient à choisir les nœuds qui ont un faible nombre de voisins tandis que minimiser la valeur Δ_2 de revient à choisir les nœuds qui sont les plus proches de la SB. Enfin, minimiser Δ_3 revient à choisir les nœuds qui ont consommés le moins d'énergie depuis le début de la mise en place du réseau.

2. Algorithme de clustering

Nous modélisons un RCSF par un graphe $G = (V, E)$ où V représente l'ensemble des nœuds et E représente l'ensemble de liens uv entre chaque paire de nœud voisins u et v . Nous utilisons les notations suivantes :

- $N(u)$, le voisinage du nœud u

- $m(u)$, la métrique du nœud u comme calculée dans l'équation (1)
- $CH(u)$, l'identité du cluster-head du nœud u
- $mCH(u)$, la métrique du cluster-head du nœud u
- $L(u)$, la liste des nœuds voisins ayant choisi u comme CH (pour chaque $v \in N(u)$, $v \in L(u)$ si $CH(v) = u$)

L'algorithme permettant de choisir les nœuds CH se décompose en deux étapes :

Etape 1 : Initialisation du nœud u dans le réseau V

Pour chaque nœud u dans le réseau V , $u \in V$

- u est son propre cluster-head $\Rightarrow CH(u) \leftarrow u$
- $m(u)$ est égale à la métrique du cluster-head de $u \Rightarrow mCH(u) \leftarrow m(u)$.
- Aucun nœud voisin v ne choisit u comme CH $\Rightarrow L(u) = \{\emptyset\}$

Etape 3 : Le nœud u est choisi comme CH par le nœud v si et seulement si :

Pour chaque nœud v dans le voisinage du nœud u , $v \in N(u)$

- la métrique $m(u)$ est inférieure à la métrique de $v \Rightarrow m(u) < m(v)$.
- la métrique de u est inférieure à la métrique du CH de $v \Rightarrow m(u) < mCH(v)$.
- le nœud v n'a pas été choisi comme CH par un autre nœud t , ($\forall t \in N(v) - \{u\}, CH(t) \neq v$)

Finalement le nœud u devient CH si et seulement si :

- il existe au moins un nœud v qui a choisi u comme CH ($\forall v \in N(u), \exists v \text{ tel que } CH(v) = u$)
- ou bien le nœud u est resté son propre CH $\Rightarrow CH(u) = u$

3. Exemple d'illustration

Afin d'illustrer le fonctionnement de notre algorithme, considérons le scénario de la figure 1. Un trait entre deux nœuds signifie que ces nœuds sont voisins. Les métriques des différents nœuds ainsi que l'identité de leur cluster-head sont affichés sur la figure 1. Initialement chaque nœud est son propre cluster-head $\Rightarrow \forall u \in V CH(u) = u$.

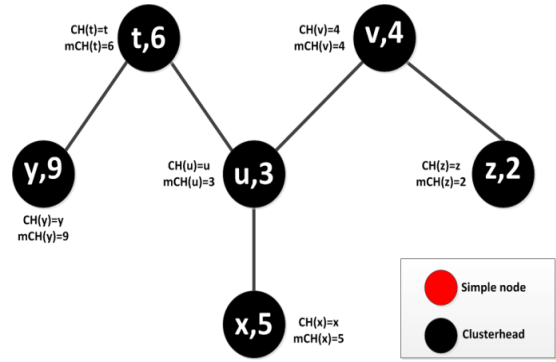


Fig. 1. Initialisation

Après exécution de l'algorithme (voir figure 2), le nœud u devient CH car il a la plus petite métrique de son voisinage. Bien que le nœud t possède dans son voisinage un nœud ayant une métrique plus faible (qui est u), le nœud t reste CH car un autre nœud notamment le nœud y l'a déjà choisi comme CH.

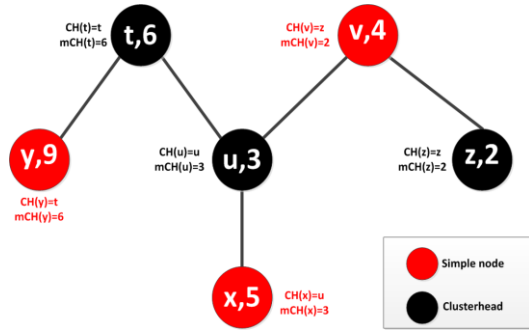
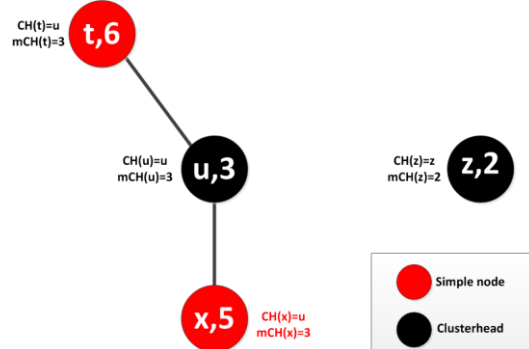


Fig. 2. Sélection de CH et formation de cluster

Après la mort des nœuds y et v , le nœud u garde son statut de CH car il a la plus petite métrique de son voisinage. Le nœud z resté également CH car c'est un nœud singleton (qui n'a pas de voisins). Le nœud t qui était précédemment le CH du nœud v , devient un nœud simple car u possède une métrique plus faible et aucun autre nœud ne l'a choisi comme CH.



Réélection des CH après la mort des nœuds y et v

IV. SIMULATIONS

Pour évaluer les performances de notre algorithme HCA, nous effectuons des simulations sous le simulateur NS2.34. Nous comparons HCA avec l'algorithme LEACH. Afin d'observer différents comportements de LEACH, nous avons

choisi d'étudier ses performances pour trois valeurs différentes du paramètre p : $p = 5\%$, $p = 10\%$.

Nous définissons la durée de vie du réseau que lorsque tous les nœuds meurent.

Le tableau 1 résume les différents paramètres de simulation.

Paramètres	Valeurs
Modèle de propagation	TwoRayGround
Couche MAC	802.15.4 CSMA-CA
Surface de simulation	500*500 m ²
Nombre de nœuds	20, 50
Taille des paquets	1Kb
Débit d'émission	256Kb/s
Energie initiale	115j
Puissance de transmission	31,32mW
Puissance de réception	35,46mW
Puissance à vide « Idle power »	0,77mW

TAB 1 : Paramètres des simulations

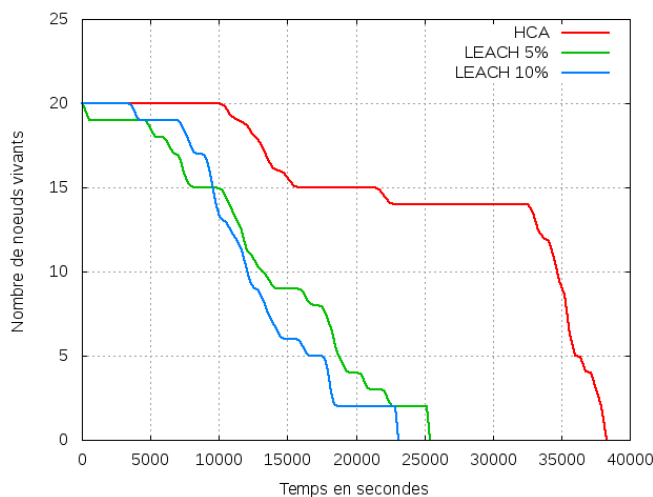


Fig. 4 : Durée de vie du réseau pour 20 nœuds

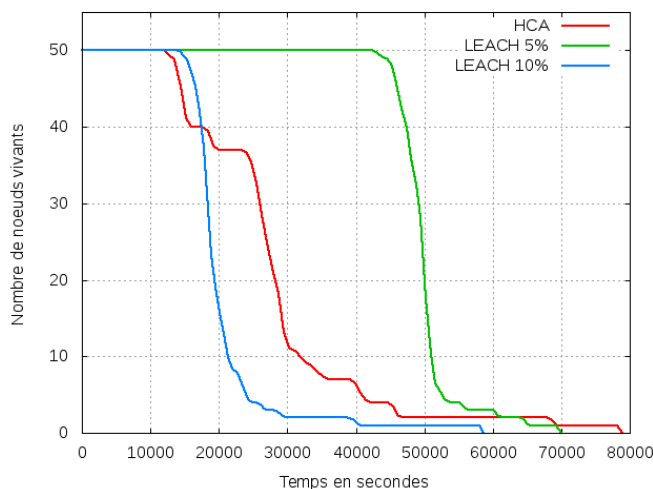


Fig. 5 : Durée de vie du réseau pour 50 nœuds

Nous définissons comme dans [17], la durée de vie du réseau comme le temps qui s'écoule avant que tous les nœuds du réseau ne meurent.

Les figures 4 et 5 illustrent la durée de vie du réseau en comparant les différents algorithmes simulés. Les résultats démontrent clairement qu'avec HCA la durée de vie du réseau est prolongée respectivement de 35% et 40% par rapport à LEACH $p=5\%$ et $p=10\%$ pour 20 nœuds. Avec 50 nœuds cette augmentation passe respectivement à 12% et 35% par rapport à LEACH $p=5\%$ et $p=10\%$. Ceci s'explique par le fait que dans HCA, lorsque les cluster-head sont hors service, ou que leur métrique a fortement augmenté, d'autres nœuds prennent leur place et ce, jusqu'à la mort de tous les nœuds. Ces résultats positifs s'expliquent par le fait que dans HCA, lorsque les cluster-head sont morts, ou que leur métrique a fortement augmenté, d'autres nœuds prennent leur place en tant que cluster-head et ce, jusqu'à la mort de tous les nœuds. Ce changement dynamique de cluster-head entraîne un *équilibre énergétique* qui permet de prolonger la durée de vie des nœuds.

V. CONCLUSIONS ET PERSPECTIVES

Dans ce papier, nous avons présenté HCA (Hybrid Clustering Algorithm), un nouvel algorithme de clustering basé sur une métrique hybride pour le choix des cluster-head dans les RCSF. Cette métrique prend en compte simultanément trois paramètres : le voisinage à un saut, la quantité d'énergie consommée et la distance des nœuds par rapport à la station de base. HCA équilibre la consommation d'énergie des nœuds dans le temps car la modification de la métrique hybride entraîne un recalcul des CH dans le temps.

Les résultats des simulations montrent clairement une augmentation de la durée de vie du réseau de l'ordre de 35 à 40% pour 20 nœuds et de 12 à 30% pour 50 nœuds.

Dans nos travaux futurs, nous prévoyons de comparer HCA avec d'autres algorithmes similaires tels que WCA [8] ou BLAC [10]. Nous prévoyons également d'affecter des poids aux différentes composantes de notre métrique hybride Δ_1 , Δ_2 et Δ_3 afin de comparer leur importance relative. Enfin, nous nous proposons de développer un routage géographique qui permettra d'utiliser au mieux les performances de HCA.

REFERENCES

- [1] Al-Karaki, J. N., Ul-Mustafa, R., & Kamal, A. E. (2004). Data aggregation in wireless sensor networks-exact and approximate algorithms. In *High Performance Switching and Routing, 2004. HPSR. 2004 Workshop on* (pp. 241-245). IEEE.
- [2] Kour, H., & Sharma, A. K. (2010). Hybrid energy efficient distributed protocol for heterogeneous wireless sensor network. *International Journal of Computer Applications*, 4(6), 0975-8887.
- [3] Hani, R. M. B., & Ijeh, A. A. (2013). A Survey on LEACH-Based Energy Aware Protocols for Wireless Sensor Networks. *Journal of Communications*, 8(3).
- [4] Johnen, C., & Nguyen, L. H. (2006). Robust self-stabilizing clustering algorithm. In *Principles of Distributed Systems* (pp. 410-424). Springer Berlin Heidelberg.
- [5] Heinzelman, W. R., Chandrakasan, A., & Balakrishnan, H. (2000, January). Energy-efficient communication protocol for wireless microsensor networks. In *System sciences, 2000. Proceedings of*

the 33rd annual Hawaii international conference on (pp. 10-pp). IEEE.

- [6] Basu, P., Khan, N., & Little, T. D. (2001, April). A mobility based metric for clustering in mobile ad hoc networks. In *Distributed computing systems workshop, 2001 international conference on* (pp. 413-418). IEEE.
- [7] Chen, H., Zhang, C., Zong, X., & Wang, C. (2013). LEACH-G: an Optimal Cluster-heads Selection Algorithm based on LEACH. *Journal of Software*, 8(10), 2660-2667.
- [8] Chatterjee, M., Das, S. K., & Turgut, D. (2002). WCA: A weighted clustering algorithm for mobile ad hoc networks. *Cluster Computing*, 5(2), 193-204.
- [9] Aissa, M., & Belghith, A. (2014). A node quality based clustering algorithm in wireless mobile Ad Hoc networks. *Procedia Computer Science*, 32, 174-181.
- [10] Ducrocq, T., Mitton, N., & Hauspie, M. (2013, April). Energy-based clustering for wireless sensor network lifetime optimization. In *Wireless Communications and Networking Conference (WCNC), 2013 IEEE* (pp. 968-973). IEEE.
- [11] Kumar, G., Mehra, H., Seth, A. R., Radhakrishnan, P., Hemavathi, N., & Sudha, S. (2014, March). An hybrid clustering algorithm for optimal clusters in Wireless sensor networks. In *Electrical, Electronics and Computer Science (SCEECS), 2014 IEEE Students' Conference on* (pp. 1-6). IEEE.
- [12] Manjeshwar, A., & Agrawal, D. P. (2001, April). TEEN: a routing protocol for enhanced efficiency in wireless sensor networks. In *Parallel and Distributed Processing Symposium, International* (Vol. 3, pp. 30189a-30189a). IEEE Computer Society.
- [13] Heinzelman, W. B., Chandrakasan, A. P., & Balakrishnan, H. (2002). An application-specific protocol architecture for wireless microsensor networks. *Wireless Communications, IEEE Transactions on*, 1(4), 660-670.
- [14] Manjeshwar, A., & APTEEN, A. D. (2002). A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks. In *Proceedings of the 2nd International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile computing* (pp. 195-202).
- [15] Lindsey, S., & Raghavendra, C. S. (2002). PEGASIS: Power-efficient gathering in sensor information systems. In *Aerospace conference proceedings, 2002. IEEE* (Vol. 3, pp. 3-1125). IEEE.
- [16] Younis, O., & Fahmy, S. (2004). HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *Mobile Computing, IEEE Transactions on*, 3(4), 366-379.
- [17] Tony DUCROCQ. Auto-organisation des réseaux sans-fil multi-sauts dans les villes intelligentes, Novembre 2013. Thèse de doctorat