



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE ABOU-BEKR BELKAID – TLEMCCEN

THÈSE LMD

Présentée à :

FACULTE DES SCIENCES – DEPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

DOCTORAT

Spécialité: *Réseaux et Systèmes Distribués*

Par :

BENMAHDI Meryem Bochra épouse HABRI

Sur le thème

Routage Intelligent dans les réseaux de capteurs à grande échelle

Soutenue publiquement le 27 Juin 2020 à Tlemcen devant le jury composé de :

Mr FEHAM Mohammed	Professeur	Université de Tlemcen	Président
M ^f LEHSAINI Mohamed	Professeur	Université de Tlemcen	Directeur de thèse
Mr GUYENNET Hervé	Professeur	Université de Besançon	Co-Directeur de thèse
Mr BOUKLI HACENE Sofiane	MCA	Université de Sidi Bel Abbes	Examinateur
Mr MERAD Boudia Omar Rafiq	MCA	Université d'Oran 1	Examinateur
Mr BENMAMMAR Badr	Professeur	Université de Tlemcen	Examinateur

*Laboratoire Systèmes et Technologies de l'Information et de la Communication (STIC)
BP 119, 13000 Tlemcen - Algérie*

Remerciements

Avant tout, Louanges à Dieu pour l'accomplissement de cette thèse.

J'ai eu de la chance de travailler avec deux directeurs de thèse remarquables, non seulement pour leurs qualités scientifiques, mais aussi pour leurs qualités humaines et leurs expériences.

J'exprime ma profonde gratitude à mon directeur de thèse Mr LEHSAINI Mohamed pour son esprit scientifique, sa pédagogie, sa disponibilité, ses conseils, et de m'avoir accordé son temps, mais aussi pour la motivation qu'il m'a su m'insuffler durant ma thèse.

Je tiens également à remercier vivement mon co-directeur de thèse Mr GUYENNET Hervé pour sa collaboration si précieuse et pour les heures de travail qu'il m'a accordé. Ses remarques pertinentes et ses conseils précieux ont largement contribué à l'aboutissement de cette thèse.

Je suis très honoré par la présence de Mr FEHAM Mohammed, qui a accepté de présider le jury de cette thèse. Je suis également très honorée par la présence de Mr BENMAMMAR Badr, Mr BOUKLI HACENE Sofiane, et Mr MERAD BOUDIA Omar Rafik qui m'ont fait l'honneur d'accepter d'être les examinateurs de cette thèse.

Qu'ils trouvent ici mes plus sincères remerciements pour l'effort qu'ils ont fourni à lire ce manuscrit et l'intérêt qu'ils ont porté à mon travail.

Je souhaite également remercier tous les membres du laboratoire STIC au sein duquel ce travail a été réalisé et mes collègues enseignants et doctorants à qui je souhaite l'épanouissement dans leur travail.

Je remercie bien évidemment tous mes collègues du vice rectorat des relations extérieures et spécialement le vice recteur professeur Abdellatif MEGNOUNIF, de m'avoir toujours soutenu.

Je souhaite aussi témoigner ma gratitude et mes vifs remerciements envers tous ceux qui ont contribué, de près ou de loin à la réalisation de ce travail.

Bien sûr, je ne peux terminer sans remercier mes proches de tout cœur et notamment ma petite famille pour tous leurs encouragements et leur amour. J'espère qu'ils me pardonneront les heures interminables que j'ai passé loin d'eux afin de réaliser cette thèse.

**"Un bien est agréable que si l'on partage"
Sénèque**

Dédicace

Je dédie ce travail

A mes très chers parents qui m'ont poussé, soutenu et épaulé pour que je puisse
atteindre mes objectifs,

A mon très cher mari qui a cru en moi et qui a toujours été là quand j'en avais besoin,

A ma petite princesse Malak qui est ma raison de vivre,

A ma sœur et à mon frère qui ont toujours été là pour moi,

A ma belle-famille qui m'a toujours encouragé et soutenu.

A mes Grand-mères qui m'ont toujours inspiré le respect et la motivation pour les
études,

Et à la mémoire de mes Grand-pères.

Un grand merci à vous tous!!

Contents

Liste des figures	iv
List of Tables	vi
Introduction Générale	1
1 Les réseaux de capteurs sans fil: Généralités et Concepts	4
1.1 Introduction	4
1.2 Les réseaux de capteurs sans fil (RCSFs)	5
1.2.1 Architecture d'un nœud de capteurs sans fil	5
1.2.2 Les caractéristiques liées aux nœuds capteurs	6
1.2.3 Architecture d'un RCSF	7
a) Les RCSFs plats	7
b) RCSFs hiérarchiques	8
1.2.4 Types des RCSFs	9
a) RCSFs terrestres	9
b) RCSFs sous-marins	10
c) Les réseaux de capteurs souterrains	11
d) Les réseaux de capteurs multimédia sans fil	13
e) Réseaux de capteurs mobiles sans fil	15
f) Les réseaux de capteurs corporels	16
1.3 Les différents facteurs de conception d'un RCSF	16
1.4 Domaines d'application des RCSFs	23
1.4.1 Applications militaires	23
1.4.2 Applications agricoles	24
1.4.3 Applications de surveillance	25
1.4.4 Applications environnementales	25
1.4.5 Applications médicales	26
1.4.6 Applications domotiques	26
1.5 Technologies de communication dans les RCSFs	27
1.5.1 Bluetooth/IEEE 802.15.4	27
1.5.2 Zigbee	27
1.5.3 Dash7-ISO/IEC 18000-7	27
1.6 Architecture protocolaire	27
1.7 Paradigme du routage dans les RCSFs	28
1.7.1 Les classes des protocoles de routage	29
a) Routage à plat	29

b)	Routage hiérarchique	30
c)	Routage basé sur la localisation	30
1.7.2	Le routage dans les RCSFs à grande échelle	30
1.7.3	Apport des protocoles de routage intelligent	31
1.8	Clustering dans les RCSFs	31
1.8.1	Principaux objectifs du clustering dans les RCSFs	32
1.8.2	Propriétés du clustering	33
1.9	Conclusion	34
2	État de l'art sur les protocoles de routage dans les réseaux de capteurs	35
2.1	Introduction	35
2.2	Taxonomie des protocoles de routage dans les RCSFs	35
2.2.1	Les protocoles de routage basiques	37
a)	Les protocoles de routage à plat	37
b)	Les protocoles de routage hiérarchiques	41
c)	Routage basé sur la localisation	46
2.2.2	Les protocoles de routage basés sur les méthodes d'optimisation	49
a)	Les protocoles de routage basés sur les métaheuristiques	49
b)	Les protocoles de routage basés sur l'apprentissage automatique	69
2.3	Conclusion	82
3	Apport du clustering sur le routage dans les RCSFs	85
3.1	Introduction	85
3.2	Préliminaires	87
3.2.1	L'approche K-Means	87
3.2.2	Méthode "Rule of Thumb"	88
3.2.3	Méthode Silhouette	88
3.2.4	Méthode Elbow	89
3.3	Travaux connexes	90
3.4	Schémas de routage proposés	91
3.4.1	Phase de configuration	91
a)	Détermination du nombre de clusters	91
b)	Formation des clusters	92
c)	Election des clusterheads	93
3.4.2	Phase de transmission de données	95
3.4.3	Complexité de l'élection des CHs	95
a)	Complexité temporelle	96
b)	Complexité en termes de messages échangés	96
3.5	Evaluation des performances	98
3.5.1	Environnement de simulations	98
3.5.2	Résultats de simulations	99
a)	Consommation d'énergie	100
b)	Durée de vie du réseau	101
c)	Nombre de paquets envoyés à la station de base	102
3.6	Conclusion	102

4 Schémas de routage basés sur le clustering pour les RCSFs à grande échelle	104
4.1 Introduction	104
4.2 Contexte	105
4.3 Travaux connexes	106
4.4 Travaux proposés	108
4.4.1 Hypothèses	109
4.4.2 Outil de simulation	109
4.4.3 Modèle d'énergie radio	109
4.4.4 Contribution 1: Schémas de routage Dist-K-Means et Imp-Dist-K-Means	110
a) Présentation de Dist-K-Means et Imp-Dist-K-Means	110
b) Évaluation et résultats de simulations	113
4.4.5 Contribution 2: Schéma de routage Cond-Dist-K-Means	115
a) Présentation de Cond-Dist-K-Means	116
b) Evaluation des performances de Cond-Dist-K-Means	117
4.4.6 Contribution 3: Schéma de routage GA-Dist-K-Means	121
a) Présentation de GA-Dist-K-Means	122
b) Evaluation et résultats de simulations	128
4.5 Récapitulatif sur les contributions proposées	129
4.6 Conclusion	131
Conclusion Générale	132
Bibliography	135

Liste des figures

1.1	Exemple d'un réseau de capteurs sans fil	5
1.2	Les composants de base d'un nœud capteur [1]	6
1.3	RCSF Plat utilisant un mode de transmission multi-saut	8
1.4	Types de RCSFs [2]	9
1.5	Exemple d'un RCSF terrestre	10
1.6	Exemple d'un RCSF sous-marin [3]	11
1.7	Exemple d'un RCSF souterrain [4]	12
1.8	Architecture d'un capteur multimédia [5]	14
1.9	Exemple de RCSFs mobiles	15
1.10	Exemple de réseaux WBSN [6]	17
1.11	Pile protocolaire	28
2.1	Classification des protocoles de routage dans les RCSFs	36
2.2	Le protocole SPIN	38
2.3	Chaînage dans PEGASIS [7]	43
2.4	Clustering hiérarchique dans TEEN et APTEEN [8]	44
2.5	Exemple de grille virtuelle dans GAF [9]	47
2.6	Organigramme des algorithmes génétiques	50
2.7	Organigramme de l'algorithme d'optimisation par PSO	55
2.8	Processus de formation de clusters dans PSO-HC	58
2.9	Comportement d'une colonie de fourmis [10]	61
2.10	Exemple d'une fonction floue pour la variable d'entrée énergie [11]	69
2.11	Un modèle de base d'un RNA	77
2.12	Étapes d'exécution de K-Means	79
3.1	Identification du point Elbow	89
3.2	Phases des schémas de routage	92
3.3	Nombre de clusters créés dans un réseau contenant 200 noeuds	93
3.4	Nombre de clusters créés dans un réseau contenant 500 noeuds	94
3.5	Communications inter-cluster et intra-cluster	96
3.6	Consommation d'énergie en fonction de la taille des réseaux	100
3.7	Durée de vie vs. Nombre de rounds	101
3.8	Nombre de paquets envoyés à la SB vs. Nombre de rounds	103
4.1	Formation des clusters par K-Means	111
4.2	Clusters générés par l'approche Imp-Dist-K-Means	112
4.3	Variation de α and β dans Imp-Dist-K-Means	114
4.4	Consommation d'énergie vs. Nombre de périodes (100 noeuds)	114

4.5	Nombre de noeuds vivants vs. Nombre de périodes (100 noeuds)	115
4.6	Consommation d'énergie vs. Nombre de périodes (200 Noeuds)	115
4.7	Nombre de noeuds vivants vs. Nombre de périodes (200 Noeuds)	116
4.8	Formation des clusters par Cond-Dist-K-means	118
4.9	Clusters générés durant la phase de re-clustering	119
4.10	Nombre de clusters formés par Cond-Dist-K-Means	120
4.11	Énergie consommée en fonction du nombre de périodes	121
4.12	Nombre de noeuds vivants en fonction du nombre de périodes	122
4.13	Organigramme de la démarche suivie	124
4.14	Organigramme de l'établissement des chemins en utilisant l'AG amélioré .	126
4.15	Sous-réseau formé de CHs et BS	126
4.16	Représentation d'un chromosome	127
4.17	Énergie consommée vs. Nombre de périodes	130
4.18	Nombre de noeuds vivants vs. Nombre de périodes	130

List of Tables

2.1	Récapitulatif sur les protocoles basés sur le clustering	83
3.1	Determination du nombre de clusters par chaque méthode	92
3.2	Paramètres de simulation des différentes méthodes de clustering	99
4.1	Paramètres de simulation (Dist-K-Means et Imp-Dist-K-Means)	113
4.2	Paramètres de simulation (Cond-Dist-K-Means)	119
4.3	Paramètres de simulation (GA-Dist-K-Means)	129
4.4	Récapitulatif sur les contributions proposées	130

Glossary

- ABC** Artificial Bee Colony. 65
- ABC-C** Cluster-based WSN routing using Artificial Bee Colony algorithm. 66
- ACA-LEACH** An Ant Colony clustering routing Algorithm. 63
- ACO** Ant Colony Optimization. 60
- ACO-C** Ant Colony Optimization for adaptive Clustering. 63
- ADC** Analog to Digital Converter. 6, 13
- AGs** Algorithmes Génétiques. 49
- AoA** Angle of Arrival. 91
- APTEEN** Adaptive Threshold sensitive Energy Efficient Sensor Network. 44
- AR** Apprentissage par Renforcement. 73
- ASLPR** A new evolutionary based application specific routing protocol. 53
- BANT** Backward ANT. 64
- BCU** Body Control Unit. 16
- Bee-Sensor-C** An Energy-Efficient and Scalable Multipath routing protocol. 67
- BeeSensor** A Bee-Inspired power aware routing protocol. 67
- BS** Base Station. 4
- BZs** Bee Zones. 67
- C.R.D.A** Clustering Routing based on Dijkstra Algorithm. 82
- CADR** Constrained Anisotropic Diffusion Routing. 40
- CAMP** Cluster Aided Multi-Path routing protocol. 45
- CDMA** Code Division Multiple Access. 42
- CH** Cluster-head. 2, 32, 41

- CHEF** Cluster Head Election mechanism using Fuzzy logic. 70
- CLIQUE** Role-free clustering with Q-learning for WSNs. 73
- CM** Cluster Member. 58
- Cond-Dist-K-Means** A Conditional Distributed K-Means Cluster-based Routing Scheme. 2, 105, 115, 133
- CPU** Central processing Unit. 13
- DBI** Davies–Bouldin. 81
- DFLC** Distributed Fuzzy Logic-based Clustering algorithm. 71
- Dist-K-Means** A Distributed approach based on K-Means. 2, 105, 133
- DoS** Denial of Service. 23
- DTN** Delay Tolerant Networking. 11
- EAGrd** Energy Aware Gradient based routing protocol. 75
- EAUCF** Fuzzy Energy-Aware Unequal Clustering algorithm. 71
- EBAB** Energy Balanced Ant-Based routing protocol. 62
- ECG** électrocardiogramme. 16
- EEDCF** Energy-Efficient Distributed Clustering algorithm based on Fuzzy approach. 73
- EEG** électro-encéphalogramme. 16
- EMG** électromyographie. 16
- ESGrd** Energy Smart Gradient based routing protocol. 75
- FBUC** Fuzzy Based Unequal Clustering for WSNs. 72
- FCH** Cluster-head election using Fuzzy Logic for WSNs. 70
- FCM** Fuzzy C-Mean. 51, 72
- FF-Ant** Flooded Forward Ant routing. 67
- FND** First Node Died. 71, 74
- GA-Dist-K-Means** A GA-based Multihop Routing Scheme using K-Means Clustering approach. 3, 105, 121, 133
- GA-EECP** GA-based Energy-Efficient Clustering Protocol. 51
- GABEEC** GA-Based Energy Efficient Clusters. 51

- GADA-LEACH** A Genetic Algorithm based Distance-Aware routing protocol. 53
- GAP** Geographic Adaptive Fidelity. 46
- GAR** An Energy Efficient GA-Based Routing protocol. 52, 108
- GEAR** Geographic and Energy Aware Routing. 47
- GFCM** Genetic algorithm Fuzzy C-Means. 51
- GGrd** Greedy Gradient based routing protocol. 75
- GPS** Global Positioning System. 18, 46
- GPSR** Greedy Perimeter Stateless Routing. 48
- HEEMP** Hybrid Energy-Efficient Multi-path routing Protocol. 45
- IACO** WSN routing method based on Improved Ant Colony Algorithm. 64
- ICT2TSK** Improved Clustering algorithm using a Type-2 Takagi-Sugeno-Kang Fuzzy Logic System. 70
- IEEABR** Improved Energy-Efficient Ant-Based Routing. 67
- Imp-Dist-K-Means** A Distributed approach based on Improved K-Means. 2, 105, 133
- IoT** Internet of Things. 4
- IP** Internet Protocol. 46
- IRP** Intelligent Routing Process. 46
- ISM** Industrial, Scientific and Medical. 21
- KEAC** K-means based Energy Aware Clustering algorithm. 80
- KM-LEACH** improved K-Means cluster-based routing scheme for WSNs. 81
- KMV-LEACH** K-medoids hybrid. 90
- LBRP** Location-Based Routing Protocol. 48
- LEACH** Low Energy Adaptive Clustering Hierarchy. 41
- LEACH-FL** improving on LEACH protocol of WSNs using Fuzzy Logic. 70
- LEACH-GA** GA-Based Energy-Efficient Adaptive Clustering protocol. 51
- LEACH-KED** Clustering routing algorithm based on energy and distance. 80
- LF** Logique Floue. 69

- LMOR** Location-based Multi-objective Optimization Routing. 49
- m-LEACH** multihop-LEACH. 62
- M2NGA** Multi-objective Two-Nested Genetic Algorithm. 52
- ML** Machine Learning. 69
- MLRC** An energy efficient Multi-Level Route-aware Clustering algorithm. 44
- MPSICA** Multi-Particle-Swarm Immune Cooperative Algorithm. 59
- MRP** A Multi-path Routing Protocol based on clustering and ACO. 64
- MRRCE** Multi-hop Routing Reducing Consumed Energy protocol. 81
- MSL** Module de Service de Localisation. 48
- MWSNs** Mobiles Wireless Sensor Networks. 15
- NBC** Nuclear Biological Chemicals. 24
- O-LEACH** Genetic algorithm based optimized leach protocol. 54
- OZEEP** Optimized Zone based Energy Efficient routing Protocol. 52
- PCH** Primary Cluster Head. 57
- PDR** Packet Delivery Rate. 59
- PEGASIS** Power-Efficient GATHERing in Sensor Information Systems. 42
- PSO** Particle Swarm Optimization. 55
- PSO-C** Energy-aware clustering for WSNs using PSO. 56
- PSO-ECHS** PSO based Energy-efficient Cluster Head Selection algorithm. 60
- PSO-HC** PSO-Hierarchical Clustering. 57
- QoS** Quality of Service. 13, 40
- QSGrd** Q-Smart Gradient based routing protocol. 74
- RCSF** Réseau de Capteurs Sans Fil. 4
- REEP** Reliable and Energy Efficient Protocol. 40
- RLBR** Reinforcement-Learning-Based Routing protocol. 75
- RNA** Réseaux de Neurones Artificiels. 76
- RSSI** Received Signal Strength Indication. 42

- SA** Simulated Annealing. 53
- SANT** Search ANT. 64
- SAR** Sequential Assignment Routing. 40
- SCH** Secondary Cluster Head. 58
- SEH-WSN** Solar Energy Harvesting Wireless Sensor Network. 25
- SEP** Stable Election Protocol. 71
- SEP-FL** Fuzzy logic approach to improve Stable Election Protocol. 71
- SIF** Swarm Intelligence based Fuzzy routing protocol for clustered WSNs. 72
- SLN** Service de Localisation des Nœuds. 48
- SMS** Short Message Service. 26
- SNR** Signal to Noise Ratio. 78
- SOM** Self Organizing Map. 76
- SPIN** Sensor Protocol for Information via Negotiation. 37
- TDMA** Time Division Multiple Access. 42
- TEEN** Threshold sensitive Energy Efficient sensor Network protocol. 43
- TPSO-CR** Two-tier Particle Swarm Optimization protocol for Clustering and Routing.
59
- TSK** Takagi, Sugeno and Kang. 73
- UWSNs** Underwater Wireless Sensor Networks. 10
- WBANs** Wireless Body Area Networks. 22
- WBSNs** Wireless Body Sensor Networks. 16
- WLAN** Wireless Local Area Networks. 21
- WMSNs** Wireless Multimedia Sensor Networks. 13
- WSN** Wireless Sensor Networks. 10
- WSS** Total within-cluster sum of square. 89
- WUSNs** Wireless Underground Sensor Networks. 11
- ZH** Zonal Head. 67

Résumé

La réduction de la consommation d'énergie et le passage à l'échelle sont des exigences clés dans les réseaux de capteurs sans fil (RCSFs), car ces réseaux sont généralement composés d'un grand nombre de capteurs sous la contrainte énergétique. Par conséquent, l'efficacité énergétique dans ce type de réseaux est considérée comme un problème critique. Une façon d'atteindre cet objectif est de minimiser le nombre de données redondantes envoyées à la station de base à travers l'approche de clustering qui est l'une des meilleures approches en termes d'efficacité énergétique dans les RCSFs à grande échelle. Dans cette thèse, nous avons proposé des solutions économes en énergie pour les RCSFs à grande échelle. Ces solutions se basent sur une amélioration de l'approche de l'apprentissage non supervisé (K-Means) et impliquent des méthodes pour déterminer le nombre de clusters approprié (Silhouette, Elbow et "Rule of Thumb"). Dans la première contribution, nous avons évalué chacune de ces méthodes dans le but de connaître l'approche la plus adéquate pour déterminer le nombre de clusters. Dans la deuxième contribution, nous avons proposé un schéma de routage basé sur une version améliorée de K-Means. La troisième contribution est un schéma de routage basé sur un clustering dynamique et la quatrième contribution est un schéma de routage qui implique "Rule of Thumb" pour déterminer le nombre de CHs, K-Means pour organiser le réseau en clusters et un algorithme génétique amélioré pour établir les chemins entre chaque CH et la station de base.

Les schémas de routage proposés ont été développés sous Matlab. Les résultats de simulations ont montré les avantages de nos solutions en termes de consommation d'énergie, de durée de vie et de passage à l'échelle comparées à d'autres schémas de routage.

Mots clés : Silhouette, Elbow, K-Means, Rule of Thumb, Clustering, RCSFs, Algorithme génétique, Routage à grande échelle.

Abstract

Reducing energy consumption and scalability are key requirements in wireless sensor networks (WSNs), as these networks are generally composed of a large number of sensors under energy constraint. Therefore, energy efficiency in this type of networks is considered a critical problem. One way to achieve this goal is to minimize the amount of redundant data sent to the base station through the clustering approach which is one of the best approaches in terms of energy efficiency in large-scale RCSFs. In this thesis, we have proposed energy-efficient solutions for large-scale RCSFs. These solutions are based on an improvement of the unsupervised learning approach (K-Means) and imply methods to determine the appropriate number of clusters (Silhouette, Elbow and "Rule of Thumb"). In the first contribution, we evaluated each of these methods in order to know the most suitable approach for determining the number of clusters. In the second contribution, we proposed a routing scheme based on an improved version of K-Means. The third contribution is a routing scheme based on dynamic clustering and the fourth contribution is a routing scheme which involves "Rule of Thumb" to determine the number of CHs, K-Means to organize the network into clusters and an improved genetic algorithm to establish the paths between each CH and the base station.

The proposed routing schemes were developed over Matlab. Simulation results have shown the benefits of our solutions in terms of power consumption, lifetime and scalability compared to other routing schemes.

Keywords: Silhouette, Elbow, K-Means, Rule of Thumb, Clustering, WSNs, Genetic algorithm, Large-scale routing.

ملخص

يعد تقليل استهلاك الطاقة وتوسيع النطاق متطلبات رئيسية في شبكات الاستشعار اللاسلكية (WSNs)، حيث تتكون هذه الشبكات بشكل عام من عدد كبير من أجهزة الاستشعار تحت قيود الطاقة. لذلك، يعتبر ترشيد استهلاك الطاقة في هذا النوع من الشبكات مشكلة حرجية. تتمثل إحدى طرق تحقيق هذا الهدف في تقليل كمية المعطيات المتكررة المرسل إلى المحطة الأساسية من خلال نهج التجميع الذي يعد واحدًا من أفضل الأساليب من حيث ترشيد استهلاك الطاقة في شبكات الاستشعار اللاسلكية المنتشرة على أوسع نطاق. في هذه الأطروحة، اقترحنا حلولاً موفرة للطاقة لهذا النوع من الشبكات. تعتمد هذه الحلول على نسخة محسنة من التعلم غير الخاضع للرقابة (K-Means) وإستعمال طرق لتحديد العدد المناسب من المجموعات (Silhouette, Elbow, "Rule of Thumb"). في المساهمة الأولى، قمنا بتقييم كل من هذه الأساليب من أجل معرفة الطريقة الأكثر ملاءمة لتحديد عدد المجموعات. في المساهمة الثانية، اقترحنا مخطط توجيه يعتمد على نسخة محسنة من K-Means. المساهمة الثالثة هي مخطط توجيه يعتمد على التجميع الديناميكي والمساهمة الرابعة عبارة عن مخطط توجيه يتضمن "Rule of Thumb" لتحديد عدد CHs و K-Means لتنظيم الشبكة في مجموعات وخوارزمية وراثية محسنة لتحديد المسارات بين كل CH والمحطة المركزية.

تم تطوير مخططات التوجيه المقترحة باستخدام محاكي Matlab. أظهرت نتائج المحاكاة فوائد حلولنا المقترحة من حيث استهلاك الطاقة وعمر هذه الشبكات وإنتشارها على نطاق أوسع مقارنة بخطة التوجيه الأخرى.

الكلمات المفتاحية: Silhouette, Elbow, K-Means, Rule of Thumb, Clustering, WSNs, خوارزمية جينية, توجيه واسع النطاق.

Introduction Générale

Introduction Générale

Contexte

L'évolution des réseaux sans fil ainsi que l'avènement d'Internet ont permis l'accès facile des utilisateurs aux différentes applications sans tenir compte de leur position géographique [12]. De plus, ces progrès ont rendu certains dispositifs moins chers, plus mobiles, plus distribués, plus puissants, plus intelligents et plus omniprésents dans la vie quotidienne du fait de leur besoin accrue. Maintenant, il est possible de construire un système embarqué de la taille d'un portefeuille avec les capacités d'un ordinateur [13]. Dans cette optique, l'émergence des réseaux de capteurs sans fil (RCSFs) constitue essentiellement la tendance la plus récente de la loi de Moore vers la miniaturisation et l'omniprésence des dispositifs informatiques [14].

Les RCSFs appartiennent à la famille des réseaux Ad-hoc avec la possibilité d'être déployés dans des endroits hostiles, critiques et difficilement accessibles par l'homme. Ces réseaux sont constitués de plusieurs nœuds capteurs déployés à grande échelle ou d'une manière dense et une ou plusieurs stations de base. Les nœuds sont capables de capter, de traiter et d'envoyer leurs données à la station de base en utilisant des protocoles de communication dédiés. Ces capteurs sont de petite taille et de bon marché. Ils sont limités en termes de calcul, de stockage, de puissance de communication et d'énergie. La diminution de leurs coûts, ainsi que la disponibilité et l'élargissement de leur gamme ont permis d'élargir le champ d'applications de ces réseaux allant du domaine militaire vers d'autres domaines comme la santé, le transport, l'agriculture, le commerce, etc.

Comme les nœuds de capteurs ont une capacité énergétique limitée et leur remplacement est difficile voire impossible, l'efficacité énergétique reste le défi principal dans ces réseaux car elle se répercute sur la durée de vie des réseaux [15]. Un protocole de communication dans les RCSFs doit non seulement minimiser la consommation d'énergie mais aussi réduire la quantité de données envoyées dans le réseau car dans certains cas le processus le plus coûteux en termes de consommation d'énergie pour un nœud capteur est la communication et en particulier la transmission [16].

Lors d'un déploiement à grande échelle, le nombre de nœuds capteurs peut atteindre des milliers, voire plus. Ceci permet d'augmenter la précision des informations et d'accroître la portée de la détection, etc. A cet effet, les RCSFs à grande échelle ont attiré l'attention de nombreux chercheurs dont le but de prolonger la durée de vie de ces réseaux. Par ailleurs, la consommation d'énergie a un effet assez important sur le passage à l'échelle dans ces réseaux. Dans la littérature, il était prouvé à travers plusieurs travaux que le clustering est parmi les approches les plus efficaces pour résoudre le problème énergétique dans les RCSFs à grande échelle.

Dans les protocoles de routage basés sur le clustering, les nœuds capteurs sont divisés

en plusieurs groupes (clusters) selon des règles bien définies. Un chef appelé "Cluster-head (CH)" est élu pour chaque cluster. A cet effet, un cluster va contenir deux types de nœuds, un CH responsable de l'agrégation et de la gestion des données alors que les autres nœuds sont responsables de la détection de leur environnement et de la collecte d'informations. En outre, les protocoles de routage hiérarchiques ont deux niveaux de communications à savoir les communications intra-cluster et les communications inter-clusters. Hormis, la limitation des nœuds dans leur puissance de communication, les derniers sont souvent obligés de transmettre leurs données en multi-sauts pour assurer la connectivité du réseau. Cependant, un protocole de routage doit avoir la possibilité de gérer la longue distance que les données détectées prennent pour les acheminer à la station de base et l'énorme quantité de surcharge du réseau pour permettre un passage à l'échelle efficace du réseau, de gérer les informations de topologie et prolonger la durée de vie du réseau [17].

Contributions

La présente thèse propose de nouvelles approches qui permettent d'acheminer les données pour satisfaire les exigences liées à l'économie de l'énergie dans les réseaux de capteurs à grande échelle. Dans cette optique, nous avons proposé une étude comparative entre les techniques les plus répandues permettant de déterminer le nombre de clusters approprié lors d'une opération de clustering à travers un schéma de routage "A Distributed approach based on K-Means (Dist-K-Means)" [18]. Ces techniques sont "Rule of Thumb" [19], Elbow [20], et Silhouette [21, 22]. Après la détermination du nombre de clusters par chacune des trois approches, le réseau sera clustérisé par une version améliorée de la méthode K-means [23]. Puis, nous avons évalué les performances de chacune et les comparer entre elles et aux protocoles de routage LEACH [24], Imp_LEACH [25] et MODLEACH [26].

En outre, nous avons proposé trois contributions. Dans la première contribution [27], nous avons proposé deux schémas de routage basés sur le clustering: "Dist-K-Means" et "A Distributed approach based on Improved K-Means (Imp-Dist-K-Means)". Le premier (*Dist-K-Means*) est basé sur la méthode K-Means et le deuxième (*Imp-Dist-K-Means*) est une version améliorée de K-Means dans ces deux schémas de routage proposés les CHs sont élus d'une manière distribuée. La version améliorée de K-means permet de générer des clusters équilibrés, ce qui permet une répartition équitable de charge entre les CHs. Les résultats de simulations ont montré que les schémas de routage proposés équilibrent la consommation d'énergie entre les CHs et améliorent considérablement la durée de vie des réseaux par rapport au protocole LEACH [24].

Comme les clusters statiques posent des problèmes d'organisation de topologie. Dans la deuxième contribution "A Conditional Distributed K-Means Cluster-based Routing Scheme (Cond-Dist-K-Means)" [28], nous avons opté pour un clustering dynamique. La solution proposée utilise la méthode Silhouette pour déterminer le nombre optimal de clusters et l'approche K-means pour le processus de formation des clusters. Le processus de détermination du nombre de clusters est déclenché lorsque le pourcentage des nœuds actifs diminue sous un seuil fixe (5%). Les résultats de simulations ont montré que notre proposition améliore considérablement la durée de vie des réseaux par rapport au schéma de routage Dist-K-Means et au protocole LEACH [24], en particulier lorsque le nombre de nœuds augmente (passage à l'échelle).

A l'instar que les nœuds de capteurs sont limités en terme de puissance de trans-

mission. Dans la troisième contribution "A GA-based Multihop Routing Scheme using K-Means Clustering approach (GA-Dist-K-Means)", nous avons utilisé un clustering dynamique avec une communication inter-cluster (CH-to-CH). La solution proposée utilise "Rule of Thumb" pour déterminer le nombre optimal de clusters, l'approche K-Means pour le processus de formation des clusters et l'algorithme génétique (AG) pour trouver les meilleurs chemins inter-cluster. Dans GA-Dist-K-Means, lorsque tous les membres d'un même cluster cessent de fonctionner, le nombre de clusters est redéfini et de nouveaux clusters sont régénérés. Le processus de construction de chemins est déclenché si un des CHs change de statut. Les résultats de simulations ont montré que le schéma de routage GA-Dist-K-Means améliore considérablement la durée de vie des réseaux par rapport au schéma de routage Dist-K-Means et au protocole LEACH [24], en particulier lors du passage à l'échelle.

Organisation de la thèse

Pour ce faire, cette thèse est organisée en quatre chapitres:

- Dans le premier chapitre, nous présentons les réseaux de capteurs sans fil, les types d'architectures dans les RCSFs, les types des RCSFs, les différents facteurs de conception des RCSFs, leurs domaines d'applications, leurs technologies de communication ainsi que l'architecture protocolaire. Ensuite, nous abordons le paradigme de routage dans les RCSFs ainsi que le clustering dans ce type de réseaux.
- Le deuxième chapitre est un état de l'art sur les protocoles de routage dans les réseaux de capteurs sans fil avec des explications sur les méthodes utilisées pour leur conception. Cette étude nous a permis de tirer profit de leurs atouts et de proposer de nouvelles solutions de routage pour ce type de réseaux.
- Le troisième chapitre présente notre première contribution qui est une évaluation des performances de trois techniques ("Rule of Thumb", Silhouette et Elbow) permettant la détermination du nombre de clusters approprié. Dans cette contribution, après la détermination du nombre de clusters par chacune de ces méthodes, nous avons développé un schéma de routage sous-jacent à chacune en utilisant l'approche K-Means pour organiser le réseau en clusters et une méthode distribuée pour l'élection des CHs.
- Le quatrième chapitre expose trois contributions basées sur le clustering avec la méthode K-Means. La première génère des clusters équilibrés en termes de nombre de nœuds. La deuxième concerne le clustering dynamique (Cond-Dist-K-Means) et la troisième concerne la construction des chemins inter-cluster avec un algorithme génétique amélioré (GA-Dist-K-Means).

Finalement, une conclusion générale clôture ce manuscrit et rappelle les principales contributions tout au long de ce travail de thèse. Elle présente également les perspectives sous-jacentes aux différentes problématiques abordées dans cette thèse. Les travaux contenus dans ce document ont été publiés dans plusieurs conférences et journaux.

Chapitre 1

Les réseaux de capteurs sans fil : Généralités et Concepts

Chapter 1

Les réseaux de capteurs sans fil: Généralités et Concepts

1.1 Introduction

La croissance rapide dans le domaine des technologies de l'information et des circuits intégrés a conduit au développement de nœuds capteurs de petite taille et à moindre coût. Ces nœuds capteurs peuvent collaborer ensemble en formant un "Réseau de Capteurs Sans Fil (RCSF)".

Aujourd'hui, les RCSFs représentent une partie intégrante de l'Internet des Objets "Internet of Things (IoT)". Ils sont utilisés pour partager des données afin d'améliorer le contrôle environnemental de l'utilisateur. Ils sont généralement composés d'un grand nombre de nœuds capteurs déployés de manière ad hoc pour observer et interagir avec le monde physique. Chaque nœud se compose de quatre composants: capteurs, microcontrôleurs, alimentation et émetteurs-récepteurs. Les capteurs de l'unité de détection mesurent les paramètres physiques dans le monde réel tels que la température, la pression, l'humidité, les vibrations, le signal acoustique, l'infrarouge, le mouvement des véhicules, etc. La valeur détectée est traitée par l'unité de traitement et transmise à la station de base "Base Station (BS)" directement (en un seul saut) ou via des nœuds intermédiaires par l'unité de communication [1]. Par ailleurs, les RCSFs sont généralement utilisés dans les applications de surveillance et de suivi en temps réel telles que la surveillance militaire, l'agriculture, la gestion des catastrophes, la surveillance des soins de santé, l'automatisation de l'industrie, le contrôle des stocks, etc. Ils sont fréquemment déployés dans les régions où l'intervention humaine est difficile ou impossible. Néanmoins, la consommation d'énergie, la bande passante et la mémoire sont considérées comme un problème majeur dans la conception des RCSFs. En plus, comme les capteurs sont déployés dans un environnement hostile, il est très difficile voire impossible de remplacer ou de recharger leurs batteries [29]. Dans les RCSFs, le coût de transmission est plus élevé que le coût de détection et de traitement. Par conséquent, une stratégie de transmission de données économe en énergie est nécessaire pour transmettre les données collectées par les nœuds capteurs à une station de base distante pour prolonger la durée de vie du réseau.

Dans ce chapitre, nous donnons une brève présentation des RCSFs, leurs caractéristiques et aussi les défis rencontrés lors de la conception de protocoles de routage pour ce

type de réseaux.

1.2 Les réseaux de capteurs sans fil (RCSFs)

Un réseau de capteurs sans fil est un ensemble de petits dispositifs appelés "capteurs" qui sont dispersés dans un champ d'intérêt en vue de collecter des informations et les router directement ou via une communication multi-sauts à un nœud puissant appelé "station de base" qui est connecté à une machine via Internet ou bien par satellite, et permet aux utilisateurs distants d'accéder aux données collectées [30] comme le montre la Figure 1.1.

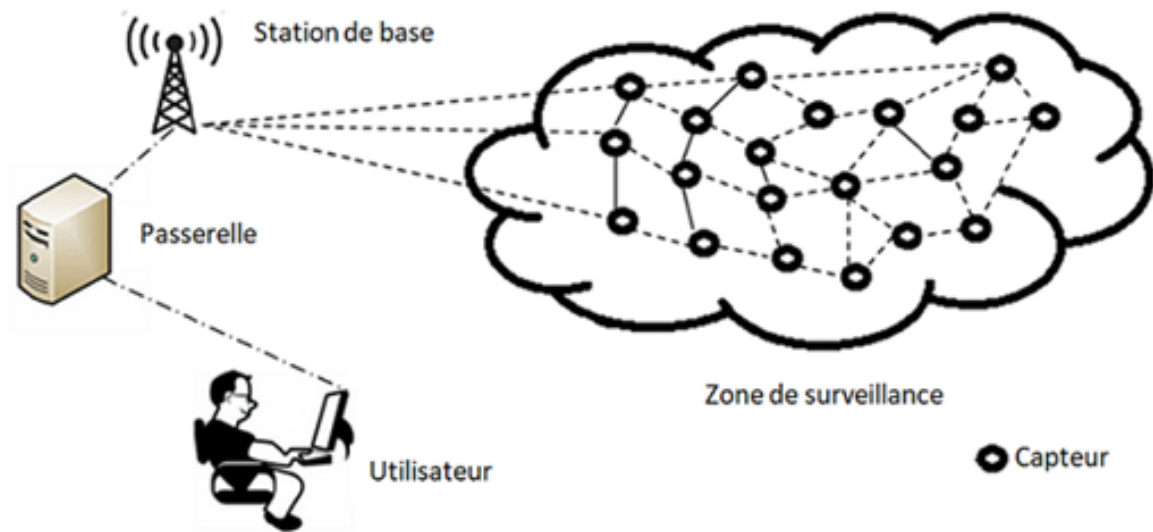


Figure 1.1: Exemple d'un réseau de capteurs sans fil

La portée de transmission des nœuds capteurs dépend de cinq facteurs: la puissance d'émission des paquets de données, la fréquence, la modulation, la localisation et les conditions météorologiques [31]. Elle peut aller de quelques mètres à des centaines de mètres. Par exemple, la portée de transmission de TelosB est de 75m à 100m en outdoor et de 20m à 30m en indoor [32]. De ce fait, en fonction de déploiement des nœuds capteurs, ces derniers ne seront pas forcément à la portée radio de la station de base. D'où, ils doivent collaborer entre eux pour envoyer les données collectées jusqu'à la station de base via un schéma de routage multi-sauts. Les conditions de déploiement vont alors poser de nouvelles contraintes et de nouveaux défis.

1.2.1 Architecture d'un nœud de capteurs sans fil

Un nœud de capteurs est un dispositif électronique capable de relever des mesures physiques dans son environnement de déploiement doté de moyens de traitement et de communication. Il est composé de quatre unités principales: unité de captage, unité de traitement, unité de transmission, et une unité d'alimentation comme le montre la Figure 1.2. Ces quatre unités sont décrites brièvement dans ce qui suit:

- **Unité d'acquisition:** Cette unité est composée de deux entités : des capteurs qui sont capables de mesurer les grandeurs physiques de leur environnement, et des

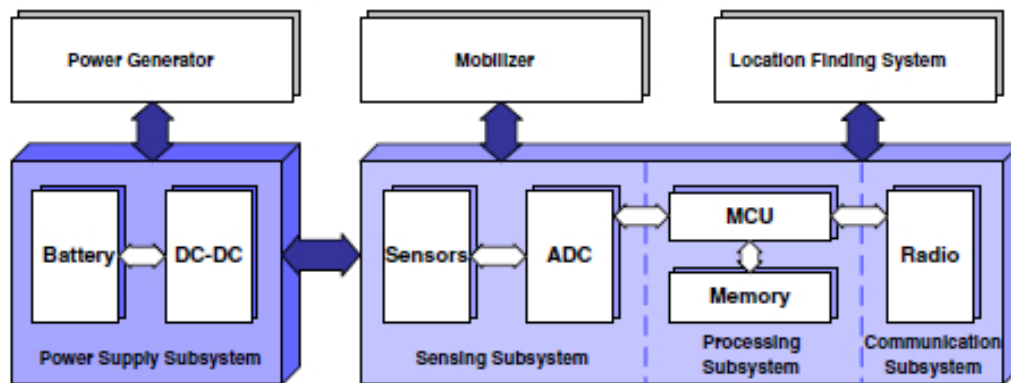


Figure 1.2: Les composants de base d'un nœud capteur [1]

convertisseurs analogique-numérique (Analog to Digital Converter (ADC)); qui sont chargés de convertir ces grandeurs collectées en signaux numériques interprétables par l'unité de traitement.

- **Unité de traitement:** Cette unité est constituée d'une mémoire interne et d'un microcontrôleur. La mémoire interne sert à sauvegarder les données collectées et les programmes de l'application, alors que le microcontrôleur sert à traiter les données et à commander les autres unités du nœud de capteurs.
- **Unité de transmission:** Cette unité est responsable de toutes les transmissions et les réceptions via un support de communication radio qui permet aux capteurs de communiquer entre eux.
- **Batterie:** Cette unité offre de l'énergie pour le fonctionnement de tous les composants du nœud de capteurs. Cette énergie peut advenir d'une batterie ou une source d'énergie renouvelable. La consommation d'énergie est un facteur critique que doit prendre en compte toutes les applications des RCSFs à cause des limitations des ressources énergétiques des nœuds. Pour résoudre le problème d'énergie, des efforts sont aboutis à la réalisation d'une unité d'alimentation dotée de panneaux solaires [33,34]. Néanmoins, le problème d'économie d'énergie persiste puisqu'on n'a pas toujours la possibilité de s'approvisionner en énergie solaire. D'où, dans presque tous les travaux de recherche on traite la problématique concernée conjointement avec la consommation d'énergie.

Par ailleurs, en fonction des applications pour lesquelles les RCSFs sont conçus, les capteurs sans fil pourraient également avoir besoin d'autres modules, comme une unité de localisation, afin d'identifier leur position géographique, ou d'un mobilisateur pour qu'ils puissent se déplacer [1]. Certaines applications pourraient aussi avoir besoin de capteurs équipés de mémoire externe qui fonctionne comme mémoire secondaire afin de stocker la quantité de données nécessaire pour l'application [16].

1.2.2 Les caractéristiques liées aux nœuds capteurs

Les capteurs ont des caractéristiques intrinsèques [35], dans ce qui suit, nous citons les plus importantes:

1. L'énergie

L'énergie est considérée comme une ressource précieuse dans les RCSFs. Elle représente la contrainte la plus importante dans la majorité des applications à base des RCSFs puisque chaque nœud est alimenté par une batterie qui n'est généralement ni rechargeable, ni remplaçable. De ce fait, toute application dédiée à fonctionner sur un nœud capteur devrait prendre en considération la contrainte d'énergie pour que ce dernier survive pour une longue durée de vie. D'où pour préserver son énergie, il devrait passer en mode actif que lorsqu'il a de l'information à transmettre ou à recevoir et éviter les transmissions et les réceptions redondantes. Par ailleurs, les communications sont les opérations les plus coûteuses en termes de consommation d'énergie [15].

2. Portée de transmission

La portée de transmission d'un nœud capteur dépend des capacités de rayonnement des antennes et la puissance du signal mis en jeu tel que si la distance entre deux nœuds est assez grande, le risque de perte de données échangées entre eux est plus probable à cause de l'affaiblissement du signal en fonction de la distance et le coût énergétique sera élevé.

3. Puissance de traitement et de stockage

En raison du coût et de miniaturisation des capteurs, généralement des microcontrôleurs sont choisis comme unité de traitement pour les capteurs [32, 36]. Par exemple, la plateforme TelosB de Crossbow est équipée d'un microcontrôleur MSP430 à 8 MHz. La puissance de stockage et de traitement des microcontrôleurs est relativement faible alors que les nœuds dans un réseau sont chargés d'accomplir plusieurs tâches telles que l'agrégation, la compression et même des fois le calcul cryptographique très complexe dans les applications de sécurité. Pour pallier à cette limitation, les nœuds doivent coopérer entre eux pour atteindre un objectif collectif.

1.2.3 Architecture d'un RCSF

L'architecture d'un RCSF joue un rôle primordial dans la qualité des performances des différents protocoles de routage de données. Elle dépend de l'application et des techniques utilisées pour la transmission des données. Il existe principalement deux types d'architectures pour les RCSFs qui sont : les RCSFs plats et les RCSFs hiérarchiques.

a) Les RCSFs plats

Dans ce type de réseaux, les capteurs peuvent communiquer avec la station de base selon deux modes: mono-saut ou multi-sauts. Dans le mode de transmission mono-saut tous les capteurs transmettent directement leurs données collectées à la station de base en utilisant une forte puissance de transmission ce qui conduit à un épuisement rapide de leurs batteries. Cependant, dans le deuxième mode, tous les capteurs coopèrent entre eux via un routage multi-sauts dans lequel chaque capteur a le même rôle. Quand un capteur veut communiquer avec une station de base se trouvant hors de sa portée de transmission, il impliquera d'autres nœuds intermédiaires comme relais. La Figure 1.3 illustre ce mode de communication avec la station de base. Par ailleurs, lors du transfert de données en utilisant une communication multi-sauts d'autres capteurs participent au routage en plus

du capteur qui a collecté la donnée et donc chaque capteur figurant dans le chemin de routage établi, consommera une quantité de son énergie pour relayer ces données. Ce qui rend le passage à l'échelle très critique et la consommation d'énergie pour le routage des données consistante.

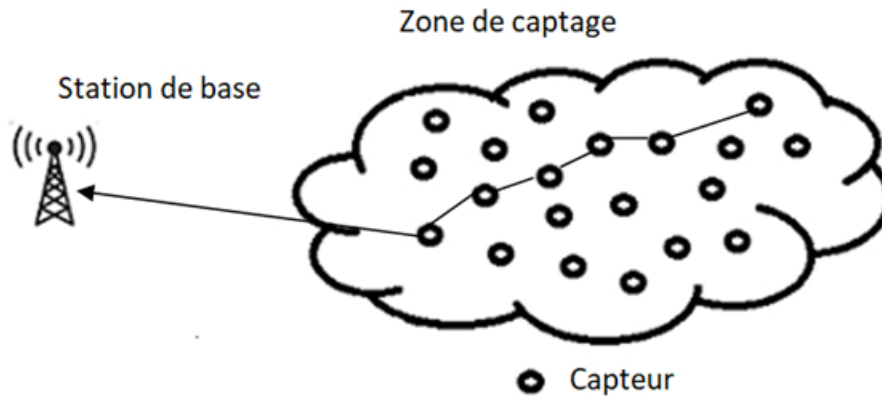


Figure 1.3: RCSF Plat utilisant un mode de transmission multi-saut

L'approche plate présente certains avantages tels que la non-nécessité de maintenir la topologie et fournit des liens de qualité de la source à la destination. Cependant, elle utilise l'approche par inondation est une opération coûteuse en termes de consommation d'énergie. De plus, elle entraîne une consommation de bande passante élevée en raison de messages redondants et une consommation d'énergie non uniforme avec un délai élevé [37].

b) RCSFs hiérarchiques

Contrairement aux réseaux présentés ci-dessus, ce type de réseaux sont des réseaux où tous les nœuds n'ont pas les mêmes rôles et peuvent ne pas avoir les mêmes ressources. Ces ressources peuvent concerner le niveau énergétique, la capacité de calcul et la capacité de stockage. Un exemple type de réseaux hiérarchiques dont les nœuds n'ont pas les mêmes rôles sont des réseaux utilisant le clustering.

Dans une approche hiérarchique, les nœuds sont regroupés en clusters et, selon certains critères, un chef de cluster est sélectionné et ce dernier sera responsable du routage de données. En outre, dans le routage hiérarchique, une approche à deux couches est généralement utilisée, où une couche est utilisée pour détecter l'environnement physique et l'autre est utilisée pour le routage. Les nœuds à faible énergie sont utilisés pour la détection tandis que les nœuds à haute énergie sont souvent utilisés pour collecter, agréger et transmettre les données [38].

L'approche de clustering est la technique la plus utilisée pour l'efficacité énergétique afin de permettre le passage à l'échelle et une communication efficace. Les approches hiérarchiques basées sur des clusters présentent certains avantages tels que le passage à l'échelle; l'agrégation efficace des données et la bonne utilisation de la bande passante du canal. Le principal problème du clustering réside dans la génération d'un clustering non uniforme qui conduit à une dissipation d'énergie élevée des nœuds, à une augmentation de la consommation totale d'énergie, et à une connectivité réseau non garantie [37, 39].

1.2.4 Types des RCSFs

Il existe plusieurs types de RCSFs qui peuvent être déployés sur terre, sous terre, sous l'eau, ou sur le corps des êtres vivants et l'utilisation de l'un de ces types dépend de leurs applications. Par ailleurs, selon l'environnement, un RCSF est confronté à différents challenges et contraintes. Néanmoins, le principal challenge pour tous les types de RCSFs reste la contrainte énergétique car le remplacement et le rechargement des batteries des noeuds qui les composent, est difficile voire impossible surtout quand ces noeuds sont déployés dans des endroits hostiles où l'intervention humaine est quasiment impossible. Nous pouvons classer les réseaux de capteurs en six classes: terrestres, corporels, souterrains, sous-marins, multimédias et mobiles comme le montre la Figure 1.4. Dans ce qui suit, nous présentons chacune de ces classes.

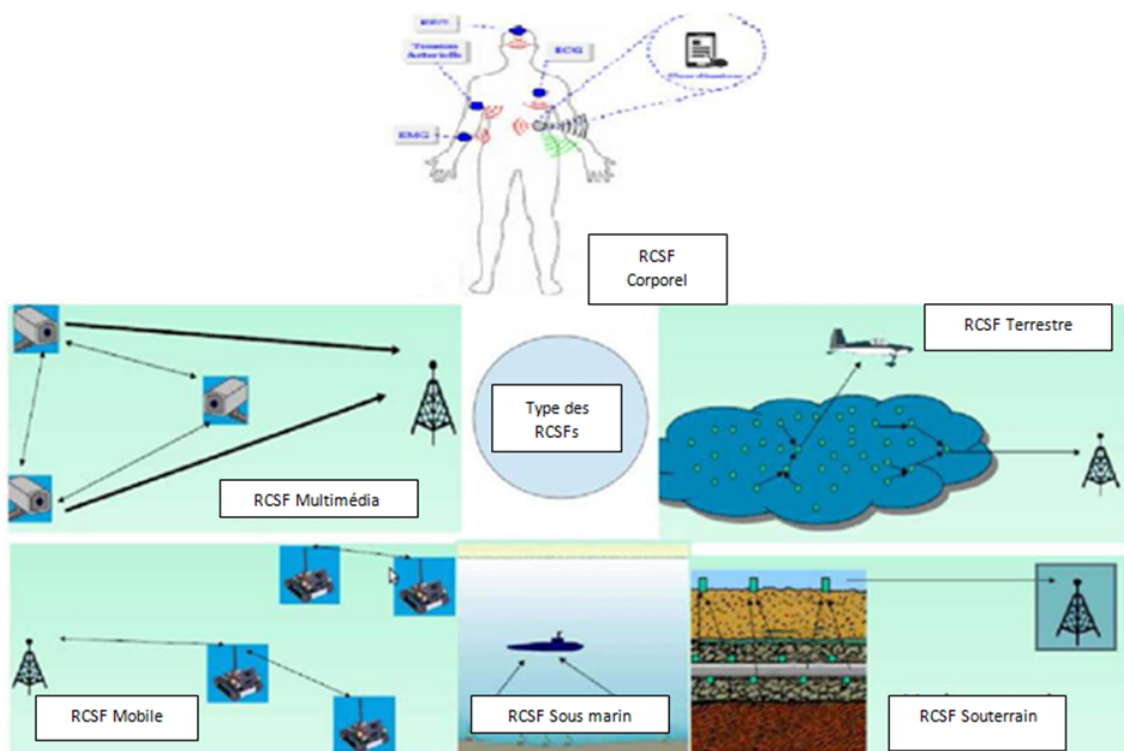


Figure 1.4: Types de RCSFs [2]

a) RCSFs terrestres

Un RCSF terrestre se compose de centaines voire de milliers de capteurs qui sont déployés aléatoirement ou d'une manière organisée dans une zone d'intérêt sur terre. Les capteurs terrestres doivent être capables de communiquer avec la station de base d'une manière efficace et fiable. Mais comme l'énergie de la batterie est limitée et ne peut être rechargeable, ces capteurs terrestres peuvent cependant être équipés d'une source d'alimentation secondaire telle que les cellules solaires [40]. Dans tous les cas, il est important de conserver l'énergie pour les nœuds capteurs. Les RCSFs terrestres peuvent être utilisés dans plusieurs domaines comme la détection et la surveillance de l'environnement, la surveillance de l'industrie et les explorations de la surface. La Figure 1.5 représente un exemple

d'un RCSF terrestre.

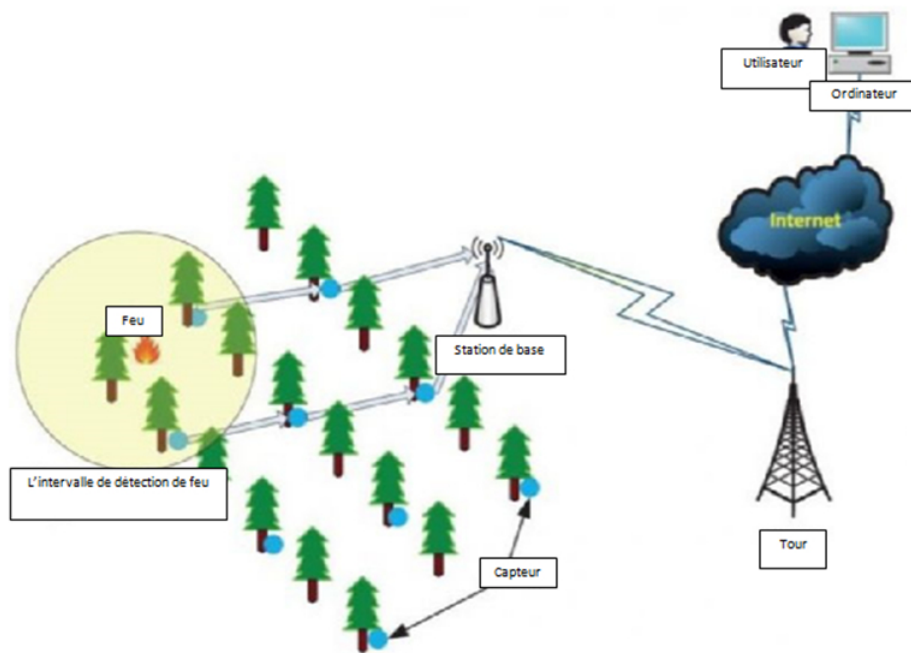


Figure 1.5: Exemple d'un RCSF terrestre

b) RCSFs sous-marins

Les réseaux de capteurs sans fil sous-marins "Underwater Wireless Sensor Networks (UWSNs)" trouvent activement leur chemin dans les recherches actuelles où la communication de bout en bout est principalement axée sur la notion de diverses perspectives. Ainsi, la modernisation des équipements et des capteurs implique des technologies de l'information et de la communication. En général, la transmission de données de la source à la destination sous l'eau est asymétrique, i.e l'émetteur et le récepteur sont actifs sur des fréquences distinctes complétant ainsi la communication sous l'eau. En outre, le routage dans les UWSNs semble être difficile, ce qui nécessite une exploration plus approfondie pour compléter la communication sans fil sous-marine de manière périodique.

Par ailleurs, la croissance rapide des mécanismes de calcul dans les UWSNs a été explorée par les chercheurs. La communication de données sur le globe terrestre dont la surface est couverte par 71% d'eau est examinée par différentes techniques. Les composants architecturaux des UWSNs sont des capteurs sous-marins qui sont déployés aléatoirement et qui collectent les informations de l'environnement sous-marin et migrent vers le nœud puits positionné à la surface de l'eau. Par la suite, ce dernier envoie les données à sa station de base (BS) correspondante. L'architecture des UWSNs est illustrée par la Figure 1.6. La gamme variée d'applications dans l'environnement sous-marin comprend la surveillance marine, la collecte de données océanographiques, l'éradication des minéraux sous-marins et la prévention des catastrophes, etc. [41]. Parmi ces applications, les connexions intermittentes jouent un rôle important en milieu sous-marin. De plus, la plupart des recherches ont suggéré que les UWSNs simulent certaines propriétés standards que les réseaux de capteurs sans fil terrestres (Wireless Sensor Networks (WSN)).

La radiofréquence (RF) est une communication de couche physique conventionnelle pour les RCSFs. Cependant, la RF ne convient pas à la transmission sous-marine en raison de l'atténuation immédiate [42].

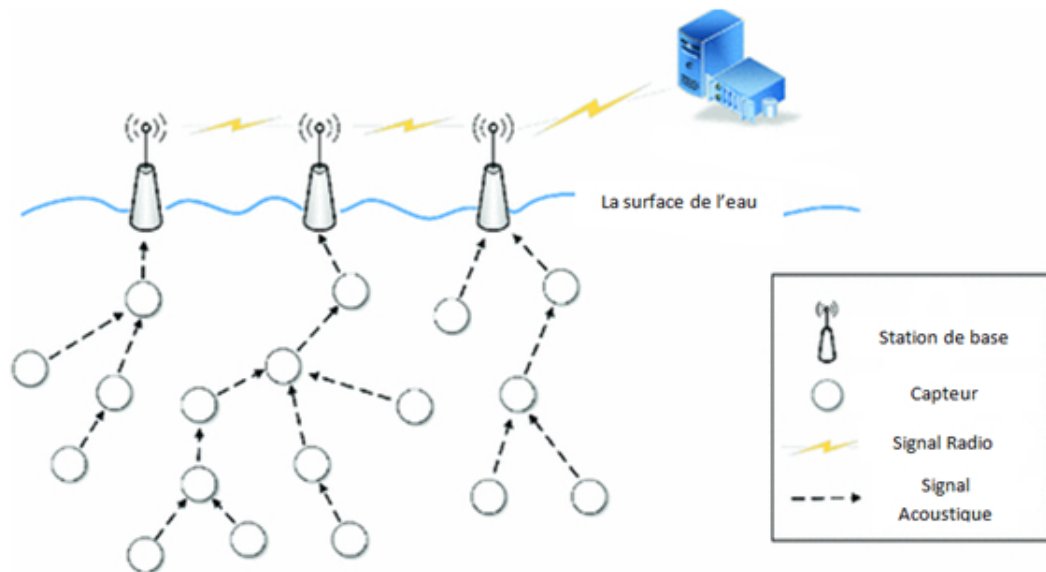


Figure 1.6: Exemple d'un RCSF sous-marin [3]

Par utilisation d'une communication optique à longue distance, la lumière peut être diffusée sur des micro-particules sous-marines. Ensuite, le signal acoustique peut se propager dans un environnement sous-marin avec une portée de 1 km et une vitesse de 1500 m/s [43]. Par ailleurs, la conception d'un protocole de routage dans un environnement sous-marin nécessite des techniques intelligentes. De plus, les UWSNs nécessitent une attention particulière pour permettre une transmission de données d'une manière fiable. Par conséquent, les recherches existantes se concentrent sur la localisation des nœuds (emplacement, profondeur et pression), la diffusion des données (unicast, multicast et flooding) et la topologie (plate et hiérarchique) basée sur des protocoles de routage. Récemment, les recherches sur les UWSNs se focalisent principalement sur la norme de support sans fil sous-marin et la communication de bout en bout doit être explorée. Pour résoudre les problèmes rencontrés ci-dessus, le protocole "Delay Tolerant Networking (DTN)" intégré à la technique JANUS qui est une méthode de signalisation acoustique, améliore le taux de livraison des paquets dans les environnements à transmission continue.

c) Les réseaux de capteurs souterrains

Les réseaux de capteurs souterrains sans fil Wireless Underground Sensor Networks (WUSNs) [29] sont composés de capteurs sans fil enfouis sous le sol. Les WUSNs constituent l'un des domaines prometteurs et permettent de nombreuses applications importantes telles que l'agriculture intelligente, diagnostic des défauts des conduites pétrolières, sauvetage en cas de catastrophe minière, exploration pétrolière et prévision des catastrophes sismiques [4]. La principale différence entre les WUSNs et les RCSFs terrestres est le milieu de communication, qui n'est plus de l'air mais du sol, de la roche ou de l'eau avec des électrolytes. Par conséquent, les capteurs enfouis ne peuvent communiquer sans fil qu'à travers le sol. La Figure 1.7 illustre un exemple de réseau souterrain.

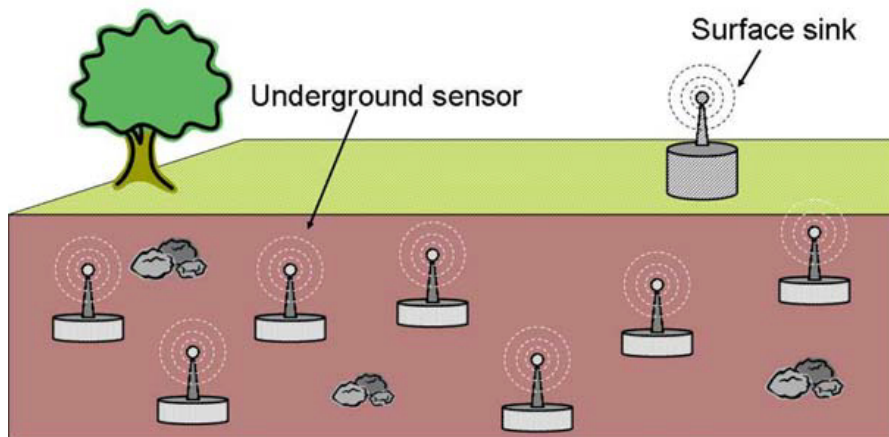


Figure 1.7: Exemple d'un RCSF souterrain [4]

Il y a eu plusieurs recherches sur l'analyse caractéristique des canaux des WUSNs. Dans [44, 45], la propagation souterraine du signal pour le système d'ondes électromagnétiques (EM) a été analysée en détails. L'effet des paramètres environnementaux tels que la teneur en eau volumétrique et des paramètres de réseau tels que la fréquence utilisée et la profondeur d'enfouissement des capteurs sur la perte de chemins et le taux d'erreurs a été mis en évidence. Dans [46], les auteurs ont étudié expérimentalement les caractéristiques de la qualité de liaison de trois canaux de communications dans les WUSNs, i.e. le canal souterrain à souterrain, le canal souterrain à hors sol et le canal hors sol à souterrain. Cependant, les résultats des travaux de recherche ont montré que l'utilisation de la propagation des ondes EM dans les WUSNs pose certains problèmes [47]. Premièrement, les ondes EM subissent des niveaux d'atténuation élevés en raison de l'absorption par le sol, les roches et l'eau souterraine. Deuxièmement, les paramètres d'environnement peuvent affecter les performances de la communication de manière imprévisible, car ces facteurs changent avec l'emplacement et varient considérablement au fil du temps. Troisièmement, les fréquences dans des plages inférieures sont nécessaires pour obtenir une plage de transmission pratique. Ainsi, par rapport à la portée de communication, la taille de l'antenne deviendra trop grande pour être déployée sous terre. Par conséquent, malgré ses applications prometteuses, le déploiement de WUSNs utilisant des techniques de propagation des ondes EM reste toujours difficile.

Ces dernières années ont vu le développement de la communication par induction magnétique (MI). Alors que les techniques utilisant les ondes EM dans un environnement souterrain rencontrent trois problèmes majeurs comme mentionné ci-dessus, l'IM est une technique alternative qui se révèle être une technologie de communication efficace pour des environnements complexes tels que le sous-sol [48]. Dans [48], l'analyse détaillée de la perte de chemins et de la bande passante du système MI dans le sol souterrain est fournie. En outre, sur la base de l'analyse du modèle de canal, des techniques de guide d'ondes MI pour la communication sont développées afin de réduire les pertes de chemins du système d'onde EM traditionnel et du système MI. De même, dans [49], un modèle de canal pour les systèmes à ondes EM et MI a été proposé. À partir de cela, le modèle multi-mode est fourni pour caractériser les canaux sans fil pour la communication sans fil dans les mines souterraines et les tunnels.

d) Les réseaux de capteurs multimédia sans fil

Les progrès technologiques incessants ont permis le développement de nouveaux types de capteurs capables de détecter et de transmettre des données multimédias (audio, images et vidéo) avec des données scalaires. Ces derniers peuvent collaborer ensemble pour former une nouvelle classe de réseaux de capteurs sans fil qui est "les Wireless Multimedia Sensor Networks (WMSNs)". Ces réseaux ont été introduits pour répondre aux exigences strictes de qualité de service "Quality of Service (QoS) des applications émergentes et pour renforcer les applications traditionnelles des RCSFs, ainsi que pour créer une série de nouvelles applications multimédias telles que la surveillance intelligente, le suivi visuel des cibles, la surveillance du trafic, la surveillance environnementale, la prestation de soins de santé avancée [5, 50].

Un capteur multimédia peut être composé de plusieurs composants de base, comme le montre la Figure 1.8: une unité de détection, une unité de traitement (Central processing Unit (CPU)), un sous-système de communication, un sous-système de coordination, une unité de stockage (mémoire) et un éventuel mobilisateur. L'unité de détection est généralement composée de deux sous-unités: les capteurs (caméras, microphones et/ou capteurs scalaires) et les convertisseurs analogique-numérique (ADC). Les signaux analogiques produits par les capteurs sur la base du phénomène observé sont convertis en signaux numériques par l'ADC, puis introduits dans l'unité de traitement. L'unité de traitement exécute l'application chargée de la coordination entre la tâche de détection et la tâche de communication et elle est interfacée avec une unité de stockage. Un sous-système de communication relie le noeud capteur au réseau et est composé d'une unité émettrice-réceptrice et d'un logiciel de communication. Ce dernier comprend une pile de protocoles de communication et des logiciels systèmes tels que des middlewares, des systèmes d'exploitation et des machines virtuelles. Un sous-système de coordination est chargé de coordonner le fonctionnement des différents périphériques réseau en effectuant des opérations telles que la synchronisation du réseau et la gestion de l'emplacement. Un mobilisateur peut permettre le mouvement ou la manipulation d'objets. Enfin, l'ensemble du système est alimenté par une unité d'alimentation qui peut être une batterie ou une unité de récupération d'énergie telle que les cellules solaires.

Les principales caractéristiques des WMSNs peuvent être décrites comme suit [5]:

- **Contraintes de ressources:** Les dispositifs de détection intégrés sont limités en termes d'énergie, de mémoire, de capacité de traitement et de débit.
- **Exigences de QoS spécifiques à l'application:** En plus des modes de livraison de données typiques des réseaux de capteurs scalaires, les données multimédias incluent des instantanés et des contenus multimédias en streaming. Les données multimédias de type instantané contiennent des observations déclenchées par un événement obtenues dans un laps de temps très court (par exemple, une image fixe). En outre, la diffusion des contenus multimédias est générée sur des périodes plus longues et doit généralement être transmis en temps réel.
- **Demande de bande passante élevée:** Les contenus multimédias, en particulier les flux vidéos, nécessitent des débits élevés. Par conséquent, les techniques de transmission pour un débit de données élevé et une faible consommation d'énergie doivent être exploitées.

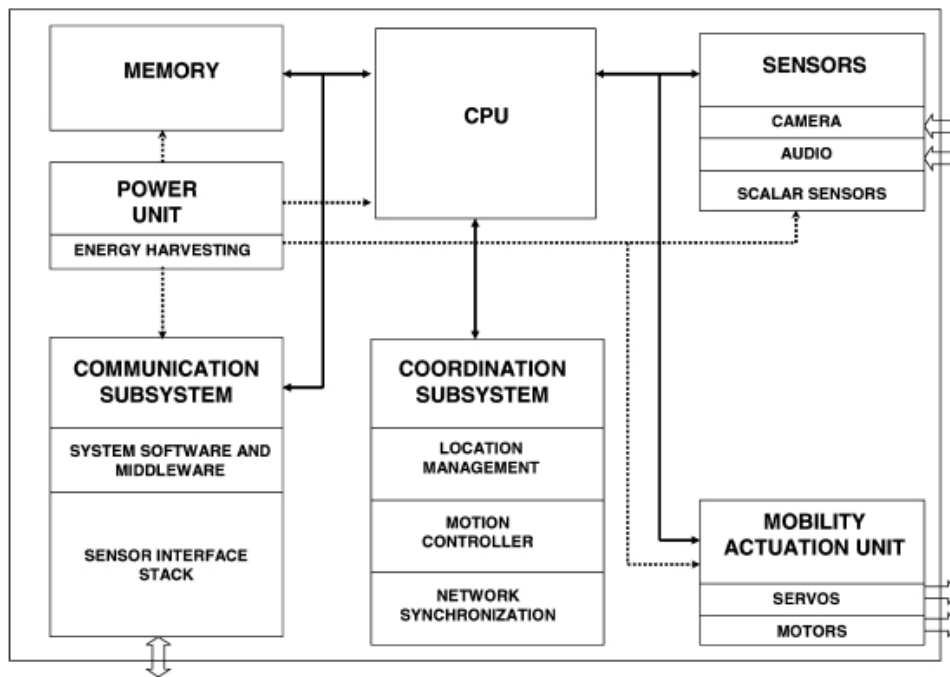


Figure 1.8: Architecture d'un capteur multimédia [5]

- **Capacité de canal variable:** La capacité et le délai pouvant être atteints sur chaque liaison dépendent de l'emplacement, varient continuellement et peuvent être de nature explosive, ce qui rend la garantie de la qualité de service (QoS) une tâche très difficile.
- **Couplage inter-couches de fonctionnalités:** En raison de la nature partagée du canal de communication sans fil, il existe une interdépendance stricte entre les fonctions gérées à toutes les couches de la pile protocolaire de communication. Cela doit être explicitement pris en compte lors de la conception de protocoles de communication visant une QoS sur les équipements à ressources limitées.
- **Techniques de codage:** Les encodeurs vidéos s'appuient sur des techniques de compression intra-trame pour réduire la redondance au sein d'une trame, et sur la compression inter-frames pour exploiter la redondance entre les trames suivantes. Étant donné que le codage prédictif nécessite des codeurs complexes, des algorithmes de traitement puissants et une consommation d'énergie élevée, il ne peut pas être adapté aux capteurs multimédias à faible coût. Cependant, il a été montré que l'équilibre traditionnel du codeur complexe et du décodeur simple peut être inversé dans le cadre du codage source distribué.
- **Traitement multimédia en réseau:** Le traitement du contenu multimédia a été principalement considéré comme un problème isolé du problème de conception du réseau. De même, la recherche qui a abordé les aspects de la livraison de contenu n'a généralement pas pris en compte les caractéristiques du contenu source et a principalement étudié les interactions entre couches parmi les couches basses de la pile protocolaire. Cependant, le traitement et la livraison de contenu multimédia

ne sont pas indépendants, et leur interaction a un impact majeur sur la QoS réalisable. La QoS requise par l'application sera fournie au moyen d'une combinaison d'optimisation inter-couches du processus de communication et de traitement en réseau des flux de données brutes qui décrivent le phénomène ciblé.

e) Réseaux de capteurs mobiles sans fil

Un réseau de capteurs mobiles sans fil (Mobiles Wireless Sensor Networks (MWSNs)) se compose de capteurs équipés de plate-formes de locomotive pour permettre leur mouvement après leur déploiement initial. Les MWSNs sont utiles pour les environnements difficiles et/ou dangereux pour le déploiement des capteurs. En plus de fonctionnalités de collecte, de traitement et de communication de données comme dans les réseaux statiques, les MWSNs permettent d'autres fonctionnalités supplémentaires. Ils peuvent se repositionner, s'organiser en réseaux et offrir une meilleure couverture en réparant les zones moins couvertes à un moment donné ainsi qu'une capacité de canal supérieure. Les défis liés aux RCSFs mobiles sont le déploiement, la localisation, l'auto-organisation, la navigation et le contrôle, la couverture, l'énergie, la maintenance et le traitement des données. Ce type de réseaux est généralement utilisé pour la surveillance de l'environnement, le suivi de cible, le sauvetage et la surveillance en temps réel des matières dangereuses. Un exemple de ce genre de RCSFs mobiles est représenté par la Figure 1.9.

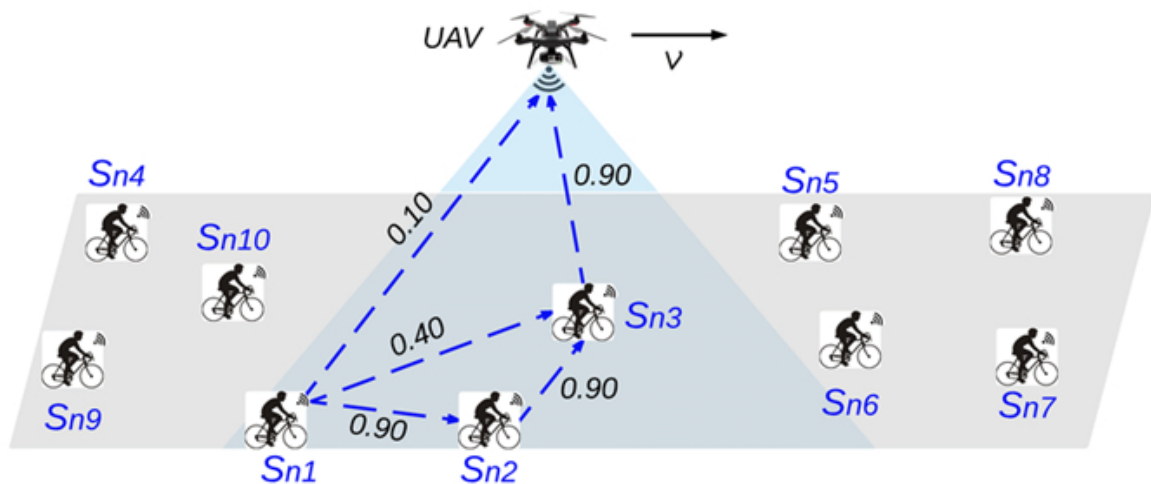


Figure 1.9: Exemple de RCSFs mobiles

Par ailleurs, en raison de la plage de détection limitée des capteurs et d'autres raisons telles que la présence d'obstacles ou l'endommagement des capteurs, il y aura toujours des zones qui ne peuvent pas être couvertes par des capteurs dans la zone d'intérêt (la zone cible), c'est-à-dire qu'il y aura inévitablement des zones qui ne sont pas couvertes (présence des trous dans la zone de couverture). Par exemple, dans le déploiement initial des réseaux de capteurs mobiles, pour des raisons environnementales ou d'autres raisons aléatoires, les capteurs sont inégalement répartis dans la zone de surveillance, et il y aura même la présence de certains endroits dans la zone de surveillance qui ne sont pas couvertes après le déploiement initial. En outre, pour diverses raisons, les capteurs peuvent présenter des anomalies de fonctionnement ou être endommagés à un instant

donné, ce qui génère des endroits non couverts par des capteurs dans la zone surveillée d'origine. Par conséquent, la présence de ces endroits non couverts rendra les données collectées par le réseau de capteurs incomplètes et entraînera une baisse de la qualité de la surveillance et même une défaillance du suivi du réseau [51].

f) Les réseaux de capteurs corporels

Un réseau de capteurs corporels sans fil "Wireless Body Sensor Networks (WBSNs)" est composé d'un ensemble de bio-capteurs communicants placés sur le corps d'un être vivant, fournissant un ensemble intégré de services personnalisés à un serveur médical. Les WBSNs permettent une surveillance physiologique des patients et sont chargés d'acheminer les données détectées (informations physiques ou vitales par exemple, glycémie, électromyographie (EMG), électrocardiogramme (ECG), électro-encéphalogramme (EEG), température, etc.) des biocapteurs au serveur médical ou non médical pour une analyse plus approfondie.

Les WBSNs sont formés de biocapteurs intelligents qui peuvent être placés à l'intérieur ou à l'extérieur de la surface du corps humain et qui permettent de détecter et accumuler les informations physiques [52, 53] du corps humain et les transmettre à serveur médical via une station de base pour un traitement ultérieur.

Dans les WBSNs, les biocapteurs peuvent être connectés via des schémas de topologie en étoile et maillée. Les données physiologiques détectées sont transmises sur le réseau à des débits différents. La glycémie, l'ECG, l'EMG, l'EEG et les biocapteurs gyroscopiques transmettent les données détectées à un débit élevé tandis que le biocapteur visuel transmet à un débit très élevé. Le biocapteur de la pression artérielle transmet à faible intensité; les biocapteurs de température transmettent à très faible débit. Le Figure 1.10 illustre un exemple de déploiement d'un réseau WBSN sur un corps humain. En se référant à cette figure, les biocapteurs hétérogènes sont placés sur le corps humain et permettent des tâches différentes. Les données collectées par ces biocapteurs sont transmises à une station de contrôle appelée Body Control Unit (BCU) et à partir de cette station elles seront transmises à un point de destination via Internet. Une fois les données envoyées, elles seront disponibles aux médecins pour une éventuelle analyse.

1.3 Les différents facteurs de conception d'un RCSF

La conception des RCSFs est influencée par plusieurs facteurs qui doivent être pris en considération.

- **Tolérance aux pannes**

La tolérance aux pannes est la capacité de maintenir les fonctionnalités du réseau de capteurs sans interruption même s'il y a des défaillances de certains capteurs [54, 55]. La fiabilité $R_k(t)$ ou la tolérance aux pannes d'un capteur est modélisée dans [55] en utilisant la distribution de Poisson pour calculer la probabilité de ne pas avoir de défaillance dans l'intervalle de temps $[0, t]$ selon l'équation suivante:

$$R_k(t) = \exp(-\lambda_k * t) \tag{1.1}$$

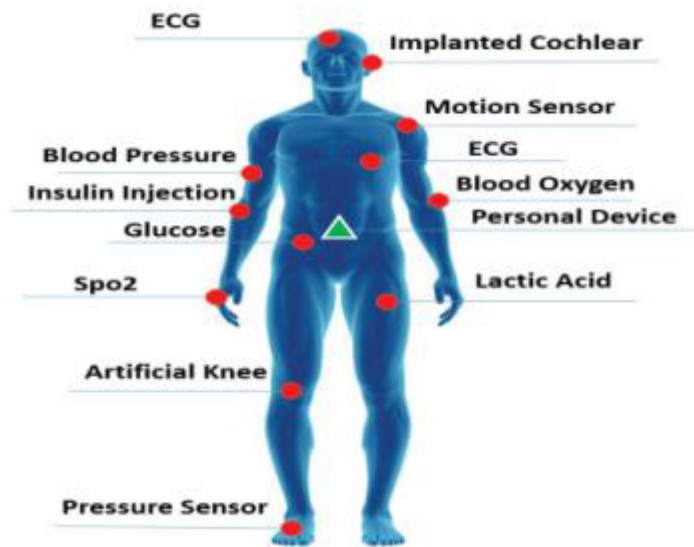


Figure 1.10: Exemple de réseaux WBSN [6]

où λ_k et t sont respectivement le taux de défaillance d'un capteur k et la période de temps.

Par ailleurs, durant le déploiement d'un réseau de capteurs ou après son déploiement, certains capteurs peuvent tomber en panne ou être bloqués en raison d'un manque d'alimentation, de dommages physiques ou d'interférences environnementales. Il faut donc penser que la défaillance des capteurs ne devrait pas affecter la tâche globale du réseau de capteurs mis en place.

Dans les RCSFs, les protocoles doivent être conçus pour répondre à un niveau de tolérance aux pannes acceptable. Si l'environnement dans lequel les capteurs sont déployés présente peu d'interférences, les protocoles peuvent être plus détendus. Par exemple, si des capteurs sont déployés dans une maison pour suivre les niveaux d'humidité et de température, l'exigence de tolérance aux pannes peut être faible, car ce type de réseaux de capteurs n'est pas facilement endommagé ou perturbé par le bruit ambiant. En outre, si des capteurs sont déployés sur un champ de batailles pour la surveillance et la détection des ennemis, la tolérance aux pannes doit être élevée car les données détectées sont critiques et les capteurs peuvent être détruits intentionnellement. Par conséquent, le niveau de tolérance aux pannes dépend de l'application des réseaux de capteurs, et les protocoles doivent être conçus dans ce contexte.

- **Passage à l'échelle**

Le nombre de capteurs déployés pour surveiller un phénomène peut être de l'ordre de centaines, de milliers ou de millions selon l'application, la superficie de la zone de déploiement et le niveau de couverture. Les nouveaux protocoles doivent pouvoir fonctionner avec ce nombre de nœuds et également prendre en considération l'aspect haute densité des réseaux de capteurs sans que les performances de ces derniers soient dégradés. La densité peut aller de quelques capteurs à quelques centaines de capteurs dans une région, qui peut être inférieure à 10 m de diamètre. La densité

peut être calculée selon l'équation 1.2 comme dans [56].

$$\mu(R) = \frac{(N * \pi * R^2)}{A} \quad (1.2)$$

où N est le nombre de capteurs déployés dans la région A ; et R la portée de transmission radio. $\mu(R)$ donne le nombre de capteurs dans le rayon de transmission de chaque capteur dans la région A .

De plus, le nombre de capteurs dans une région peut être utilisé pour indiquer la densité de capteurs. La densité des capteurs dépend de l'application dans laquelle les capteurs sont déployés. Par exemple, pour l'application de diagnostic des machines, la densité des capteurs est d'environ 300 capteurs dans une région de l'ordre de $5 \times 5m^2$, et pour l'application de suivi des véhicules cette densité est d'environ 10 capteurs par région [57]. Pour l'application de surveillance de l'habitat, le nombre de capteurs varie de 25 à 100 par région [58]. En outre, une maison intelligente peut contenir un grand nombre d'équipements dotés de capteurs et ce nombre pourra augmenter si des capteurs sont intégrés dans les meubles et autres articles. La densité sera extrêmement élevée lorsqu'une personne contenant normalement des milliers voire des millions de capteurs si ces derniers sont à l'échelle nano.

- **Coûts de production des capteurs**

Les réseaux de capteurs sont généralement composés d'un grand nombre de capteurs (des milliers ou des millions), le coût d'un seul capteur est très important pour justifier le coût global des réseaux. Si le coût du réseau est plus cher que le déploiement de capteurs traditionnels, le réseau de capteurs n'est pas justifié par les coûts. Par conséquent, le coût de chaque capteur doit être moins cher pour encourager l'utilisation de ce type de réseaux. Le développement dans la technologie sans fil a permis de réduire le coût d'un système radio Bluetooth à moins de 10 \$ [59]. En outre, Le coût d'un capteur doit être bien inférieur à 1 \$ pour que le réseau de capteurs soit réalisable [60]. Le coût d'une radio Bluetooth, qui est connu pour être un appareil à faible coût, est même 10 fois plus cher que le prix ciblé pour un capteur. De plus, un capteur peut être équipé d'un système de localisation, d'un mobilisateur ou d'un générateur d'énergie selon les applications des réseaux de capteurs. En conséquence, le coût d'un capteur est un problème très difficile étant donné le nombre de fonctionnalités avec un prix bien réduit.

- **Contraintes matérielles**

Un capteur est composé de quatre éléments de base : une unité de détection, une unité de traitement, une unité d'émetteur-récepteur et une unité de d'énergie. Un capteur peut également avoir des composants supplémentaires dépendants de l'application tels qu'un système de localisation "Global Positioning System (GPS)", un générateur d'énergie (cellule solaire) et un mobilisateur. L'unité de détection est généralement composée de deux sous-unités: des capteurs et des convertisseurs analogique-numériques (ADC). Les signaux analogiques produits par les capteurs sur la base du phénomène observé sont convertis en signaux numériques par l'ADC, puis introduits dans l'unité de traitement. L'unité de traitement, généralement associée à une petite unité de stockage, gère les procédures qui font collaborer le capteur

avec les autres capteurs pour réaliser les tâches de détection. Une unité émettrice-réceptrice connecte le capteur au réseau. L'un des composants les plus importants d'un capteur est le bloc d'alimentation. L'unité d'énergie peut être supportée par une unité de récupération d'énergie telle que des cellules solaires. Il existe également d'autres sous-unités, qui dépendent de l'application.

La plupart des schémas de routage conçus pour les RCSFs et des tâches de détection nécessitent la connaissance de l'emplacement avec une grande précision pour faciliter l'intervention dans certaines applications. Ainsi, il est courant qu'un capteur possède un système de localisation. Un mobilisateur peut parfois être nécessaire pour déplacer les capteurs lorsqu'il est nécessaire pour effectuer les tâches assignées à ces derniers. Toutes ces sous-unités peuvent avoir besoin de s'insérer dans un module de petite taille [61]. La taille requise peut être plus petite que même un centimètre cube [62], ce qui est suffisamment léger pour rester suspendu dans l'air. Outre la taille, il existe également d'autres contraintes strictes pour les capteurs. Ces capteurs doivent [63]:

- consommer une quantité d'énergie extrêmement faible,
- opérer dans des densités volumétriques élevées,
- avoir un faible coût de production et être dispensable,
- être autonome et fonctionner sans intervention humaine,
- être adaptatif avec l'environnement (sous terre, aquatique, etc.).

L'unité émettrice-réceptrice des capteurs (transceiver) peut être un dispositif optique passif ou actif comme dans les capteurs intelligents [62] ou un dispositif radiofréquence (RF). Les communications RF nécessitent des circuits de modulation, de bande passante, de filtrage, de démodulation et de multiplexage, ce qui les rend plus complexes et plus chers. Néanmoins, la communication RF est préférée dans la plupart des projets de recherche en cours sur les réseaux de capteurs, car les paquets transportés dans les réseaux de capteurs sont petits, les débits de données sont faibles, et la réutilisation des fréquences est élevée en raison de courtes distances de communication. Ces caractéristiques permettent également d'utiliser une radioélectronique à faible rapport cyclique pour les réseaux de capteurs. Cependant, la conception de circuits radio à faible consommation d'énergie et à faible rapport cyclique reste techniquement plus ou moins difficile, et les technologies radio actuelles telles que celles utilisées dans Bluetooth ou Zigbee ne sont pas assez efficaces pour les réseaux de capteurs car le basculement de l'état "allumé" à l'état "éteint" consomme beaucoup d'énergie [57].

- **Topologie des RCSFs**

Le grand nombre de capteurs qui sont déployés dans des zones dont l'accès est difficile et qui sont sujets à des pannes fréquentes, rend la maintenance de la topologie difficile. En outre, leur déploiement en grand nombre de manière dense nécessite une gestion soignée de la maintenance de la topologie. Dans ce qui suit, nous examinons les problèmes liés à la maintenance et au changement de la topologie en trois phases:

1. Pré-déploiement et phase de déploiement

Les capteurs peuvent être jetés en masse ou placés un par un dans la zone d'intérêt. Ils peuvent être déployés:

- larguer par un avion
- placer un par un par un être humain ou un robot.

Bien que le grand nombre de capteurs et leur déploiement sans surveillance empêchent généralement de les placer selon un plan de déploiement soigneusement conçu, les schémas de déploiement initial doivent :

- réduire le coût d'installation,
- promouvoir l'auto-organisation et la tolérance aux fautes.

2. Phase de post-déploiement

Après le déploiement, les changements de topologie sont dûs à des changements dans les capteurs [61]:

- leurs positions dans le cas des réseaux de capteurs mobiles,
- accessibilité due à la présence des obstacles,
- énergie disponible (les capteurs peuvent cesser de fonctionner quand ils épuisent leurs batteries),
- dysfonctionnement (à cause des pannes ou endommagement intentionnel ou par des animaux).

Les capteurs peuvent être déployés statiquement. Cependant, la défaillance de ces derniers reste inévitable à cause de l'épuisement de l'énergie ou l'écrasement par des animaux, etc.. Il est également possible d'avoir des RCSFs avec des capteurs mobiles. Par conséquent, la topologie des RCSFs est sujette à plusieurs modifications après le déploiement.

3. Phase de redéploiement de nouveaux capteurs

De nouveaux capteurs peuvent être redéployés en tout moment pour remplacer les capteurs défectueux ou en raison de changements dans la dynamique des tâches. L'ajout de nouveaux capteurs nécessite de réorganiser le réseau de nouveau. De ce fait, pour faire face aux changements fréquents de la topologie dans les RCSFs dont les capteurs ayant des contraintes de consommation d'énergie très strictes cela nécessite des protocoles de routage bien particuliers.

• **Environnement**

Les capteurs sont généralement déployés de manière dense soit très près, soit directement à l'intérieur du phénomène à observer. En plus, ils peuvent fonctionner sans intervention humaine lorsqu'ils sont déployés dans des zones géographiques éloignées. Ils peuvent être déployés :

- à l'intérieur d'une grande machine,
- au fond d'un océan,
- dans un champ de bataille au-delà des lignes ennemies,
- dans une maison ou un grand immeuble,

- attaché aux animaux, et
- attaché aux véhicules en mouvement rapide.

Ces types d'environnement dans lesquels les capteurs pourraient fonctionner, doivent être pris en compte lors de la conception des protocoles pour les RCSFs.

- **Supports de transmission**

Dans un réseau de capteurs dont la stratégie de routage est basé sur un schéma multi-sauts, les capteurs communiquent entre eux via un médium sans fil. Ces liens peuvent être formés par des supports radio, infrarouges ou optiques. Ainsi, pour permettre l'exploitation globale de ces réseaux, le support de transmission choisi doit être disponible dans le monde entier.

Une option pour les liaisons radio est l'utilisation de bandes industrielles, scientifiques et médicales "Industrial, Scientific and Medical (ISM)", qui offrent une communication sans licence dans la plupart des pays. Certaines de ces bandes de fréquences sont déjà utilisées pour la communication dans les systèmes téléphoniques sans fil et les réseaux locaux sans fil "Wireless Local Area Networks (WLAN)". Pour les réseaux de capteurs, un émetteur-récepteur de petite taille, à faible coût et à très faible puissance est requis. D'après les auteurs dans [64], certaines contraintes matérielles et le compromis entre l'efficacité de l'antenne et la consommation d'énergie limitent le choix d'une fréquence porteuse pour de tels émetteurs-récepteurs à la gamme ultra-haute fréquence. Ils proposent également l'utilisation de la bande ISM 433 MHz en Europe et de la bande ISM 915 MHz en Amérique du Nord. Les problèmes de conception de l'émetteur-récepteur dans ces deux bandes sont traités dans [65]. Par ailleurs, les principaux avantages de l'utilisation des bandes ISM sont la radio gratuite, une énorme allocation de spectre et une disponibilité mondiale. En plus, ces bandes de fréquences ne sont pas liés à une norme particulière, ce qui donne plus de liberté pour la mise en œuvre de stratégies d'économie d'énergie dans les réseaux de capteurs.

Une grande partie du matériel actuel pour les capteurs est basée sur la conception de circuits RF. Le capteur sans fil μ AMPS, décrit dans [57], utilise un émetteur-récepteur compatible Bluetooth 2.4 GHz avec un synthétiseur de fréquence intégré. Le capteur de faible puissance décrit dans [66] utilise un émetteur-récepteur RF monocanal fonctionnant à 916 MHz. Un autre mode possible de communication entre les capteurs dans les RCSFs est l'infrarouge. La communication infrarouge est sans licence et robuste aux interférences des appareils électriques. Les émetteurs-récepteurs infrarouges sont moins chers et plus faciles à construire. Cependant, l'inconvénient principal de l'infrarouge est l'exigence d'une ligne de visée entre l'expéditeur et le récepteur. Cela fait de l'infrarouge un choix non recommandé pour le support de transmission dans les RCSFs.

Les exigences d'applications inhabituelles des RCSFs rendent le choix des supports de transmission plus difficile. Par exemple, les applications marines peuvent nécessiter l'utilisation du milieu de transmission aqueux. Dans ce type de scénario, on aimerait utiliser un rayonnement de grande longueur d'onde qui peut pénétrer la surface de l'eau. Les applications dans le champ de batailles peuvent rencontrer des canaux sujets aux erreurs et des interférences plus importantes. De plus, une

antenne dans un capteur peut ne pas avoir la hauteur et la puissance de rayonnement de celles des autres appareils sans fil. Par conséquent, le choix du support de transmission doit être soutenu par des schémas de modulation robustes qui tiennent compte du milieu de déploiement des RCSFs.

- **Consommation d'énergie**

La plupart des capteurs ne peuvent être équipés que d'une batterie dont la capacité énergétique est limitée ($<0,5$ Ah, 1,2 V). Dans certains scénarios d'applications, le remplacement et/ou le rechargement de ces batteries est quasiment impossible. D'où la durée de vie du capteur dépend de la durée de vie de la batterie. Dans un réseau de capteurs où le schéma de routage est à sauts multiples, chaque capteur joue un double rôle : collecteur et relayeur de données. Le dysfonctionnement de quelques capteurs peut entraîner des changements topologiques importants et peut nécessiter une réorganisation du réseau. Par conséquent, la conservation et la gestion de l'énergie prennent plus d'importance dans les RCSFs. C'est pour ces raisons que les chercheurs donnent toujours de l'importance à la conception de protocoles qui sont économes en énergie pour les réseaux de capteurs.

Par ailleurs, la tâche principale d'un capteur dans une zone d'intérêt est de détecter des événements, d'effectuer un traitement local rapide sur les données collectées, puis de les transmettre à une station de base distante directement ou via selon un schéma de routage à sauts multiples. D'où la consommation d'énergie peut donc être divisée en trois parties: détection, communication et traitement des données. Néanmoins, l'opération de communication de données est l'opération qui consomme plus d'énergie comparée aux autres opérations [15].

- **Aspect sécurité**

Les RCSFs sont sujets à un grand nombre d'attaques et de menaces. Ils sont souvent ouverts à plusieurs menaces et intrusions externes, qui pourraient les attaquer. Ainsi, les problèmes de sécurité et de confidentialité doivent être très bien traités, l'attaquant peut cibler la disponibilité d'un RCSF en capturant ou en neutralisant un capteur particulier, ce qui entraîne parfois des anomalies dans le fonctionnement du réseau mis en place. Par exemple, dans un réseau de capteurs corporels il y a des attaques qui peuvent coûter la vie des patients. Dans ce type de réseaux, l'adversaire peut capturer ou neutraliser un capteur EEG et envoyer les fausses informations au médecin. Cela pourrait entraîner une situation potentiellement mortelle ou même causer la mort du patient. Un adversaire peut également utiliser le brouillage et l'altération. Le brouillage (interférence de radiofréquence) peut être utilisé par un adversaire sur quelques nœuds pour bloquer l'ensemble du réseau [67]. Cette attaque ne peut pas bloquer les grands réseaux, mais comme les "Wireless Body Area Networks (WBANs)" sont généralement de petits réseaux, donc elle pourra les bloquer et même entraîner une perte de paquets de données. En fait, un adversaire altère parfois physiquement les WBANs. Il est possible qu'un attaquant puisse interférer, endommager ou supplanter un réseau WBAN pour acquérir les informations personnelles sur la santé d'un patient. Il peut également utiliser une technique d'inondation pour épuiser la mémoire en envoyant à plusieurs reprises des paquets supplémentaires inutiles, que le système est incapable de les gérer. Cela empêche les

utilisateurs légitimes du réseau d'accéder aux services ou aux ressources [68]. Ceci peut se faire par une attaque par déni de service "Denial of Service (DoS)" qui vise non seulement à perturber et détruire le réseau, mais aussi à diminuer la capacité du réseau à fournir les services d'urgence nécessaires.

1.4 Domaines d'application des RCSFs

Les RCSFs peuvent être constitués de nombreux types de capteurs tels que sismique, thermique, visuel, multimédia, aquatique et acoustique, qui sont capables de surveiller une grande variété de conditions ambiantes, notamment:

- Température,
- humidité,
- mouvement des véhicules,
- pression,
- la présence ou l'absence de certains types d'objets,
- niveaux de contraintes mécaniques sur les objets attachés, et
- les grandeurs physiologiques.

Les capteurs peuvent être utilisés pour la surveillance continue, la détection d'événements spéciaux, la détection d'emplacement et le contrôle local des actionneurs. Le concept de détection et de connexion sans fil de ce type de noeuds promet de nombreux nouveaux domaines d'applications. Nous pouvons classer les applications dans les domaines suivants: militaire, environnemental, médical, et domestique. Il est possible d'élargir cette classification avec d'autres catégories telles que l'exploration spatiale, le traitement chimique et les secours en cas des catastrophes.

1.4.1 Applications militaires

Les RCSFs sont devenus un outil essentiel pour les applications militaires impliquant la détection d'intrusions, la surveillance de divers paramètres, la collecte d'informations et le soutien logistique intelligent dans une zone de déploiement inconnue. Ces réseaux peuvent fournir différents services aux forces armées et aériennes comme la collecte d'informations, la surveillance des champs de bataille et la détection d'attaques. En raison de leurs capacités de transmission en temps réel, les RCSFs jouent un rôle important dans les opérations militaires. Ces réseaux offrent plusieurs avantages par rapport aux dispositifs de détection traditionnels tels que la tolérance aux pannes, la robustesse et le déploiement à faible coût. En cas d'une attaque, certains capteurs peuvent être endommagés mais les dommages d'un certain nombre de capteurs dans les RCSFs ne doivent pas perturber le réseau tout entier si le concept de tolérance de pannes a été pris en considération lors de la mise en place de ce type de réseaux.

Il existe diverses études sur la mise en œuvre des RCSFs dans les applications militaires. Ces applications concernent la détection des intrusions, le suivi de l'ennemi, la surveillance du champ de batailles, et la détection des attaques de produits chimiques biologiques nucléaires "Nuclear Biological Chemicals (NBC)" [69, 70].

1. Détection d'intrusions: Les réseaux de capteurs peuvent être utilisés comme un système de détection d'intrusions. Ainsi, au lieu d'utiliser des mines, l'intrusion peut être repérée à l'aide d'un réseau de capteurs dans une zone particulière. Les mines [71, 72] peuvent cibler les civils aussi bien qu'une alternative aux mines; les capteurs peuvent détecter l'environnement et alarment les forces chargées de la surveillance de la zone concernée.
2. Surveillance du champ de batailles: Les zones critiques et les frontières peuvent être étroitement surveillées à l'aide des RCSFs pour obtenir des informations sur toute activité ennemie dans cette zone. Cela permet une collecte rapide des informations et un délai de réponse très court [73].
3. Charge acoustique de l'aérostat pour la détection des hélicoptères: l'utilisation de détecteurs acoustiques suspendus sous des aérostats a permis de localiser et limiter les mouvements transitoires des mortiers, des gros canons et des petits bras. Le même aérostat et le même capteur de sol sans surveillance "UGS" peuvent suivre des hélicoptères adjacents en 3D pour éviter les évasions [74].
4. Un nouveau scanner de chaussures utilisant un capteur de résonance: Ce scanner de chaussures a été développé par le centre d'excellence "GE Security" de San Diego. Ce scanner utilise la quadruple résonance (QR) pour distinguer les explosifs rangés dans les chaussures [75].

Des avancées dans les applications de filtrage de groupes ont permis le développement de capteurs pour la détection des mines marines, un système miniaturisé de capteurs pour les opérations de surveillance jour/nuit, des capteurs acoustiques à faible coût pour les opérations anti-sous-marins littoraux.

5. Applications de filtrage de groupes pour la détection des mines marines: les applications développées pour les forces marines doivent prendre en compte la spécificité du milieu sous-marin. Par exemple l'application "Naval Surface Warfare Center de Panama City (NSWC PC)" utilise des capteurs de véhicules sous-marins sans pilote (UUV) pour produire l'image du sonar et caractériser les propriétés de l'image géométriquement [76].

1.4.2 Applications agricoles

Dans l'agriculture intelligente, les technologies modernes de détection, de calcul et de communication telles que l'Internet des objets (IoT) sont utilisées dans les applications agricoles existantes. Les avantages d'une agriculture intelligente sont d'augmenter la productivité, la qualité de la production agricole, de gérer et de contrôler efficacement les systèmes agricoles et, par conséquent, d'augmenter les revenus des personnes. Les objets connectés comprenant des nœuds de RCSF sont largement utilisés pour la surveillance et le contrôle de la température [77], de la lumière [78], de l'arrosage intelligent et de

l'agriculture intelligente [79]. Le problème majeur avec la conception des RCSFs est que l'énergie de la batterie du capteur est limitée et est consommée en peu de temps en fonction du cycle de service de l'application. Une nouvelle idée pour augmenter la durée de vie des réseaux de capteurs consiste à utiliser des sources d'énergie renouvelables telles que l'énergie solaire photovoltaïque. Les modifications apportées à la norme IEEE 802.15.4 (appelée aussi ZigBee) pour la communication sans fil à faible débit dans la norme ZigBee Pro par les alliances ZigBee, permettent la fonction de récupération d'énergie dans les RCSFs traditionnels. Cependant, l'énergie solaire n'est pas constante et a un caractère très intermittent en particulier dans les pays qui ont moins de soleil durant l'année tels que les pays de l'Europe. Par conséquent, même si le système WSN "Solar Energy Harvesting Wireless Sensor Network (SEH-WSN)" de récupération d'énergie solaire conçu de manière optimale est déployé, les variations de l'irradiance solaire peuvent dégrader la durée de vie des réseaux.

Il existe de nombreuses opérations de surveillance et de contrôle dans les domaines agricoles comme la mesure du niveau d'humidité dans le sol et les plantes, l'arrosage des plantes, la distribution de pesticides, la gestion des déchets, le contrôle des animaux, etc. Par exemple, dans un scénario de mesure du niveau d'humidité, si les capteurs détectent le niveau d'humidité dans le sol descend au-dessous d'un certain seuil prédéfini, ils envoient un message d'alerte à un poste de contrôle distant via une passerelle sans fil connectée à Internet [80]. Par suite, selon les valeurs de seuil prédéfinies envoyées, les actionneurs nécessaires peuvent être activés comme le contrôle de la vitesse de la pompe à eau, la mise en marche/arrêt du moteur de la pompe à eau, la récolte des cultures, l'ensemencement des grains, la distribution des pesticides sur les cultures, etc. [81].

1.4.3 Applications de surveillance

La surveillance par des capteurs a connu un gain significatif en termes de sécurité et de financement. Dans ce type d'applications, les capteurs peuvent être placés dans des endroits sensibles qui ne sont pas accessibles par des êtres humains, et en cas d'intrusion dans une zone de surveillance, ils remontent immédiatement des alertes au centre de contrôle. Par exemple dans les maisons dites "maisons intelligentes", il y aurait plusieurs types de capteurs placés à des endroits différents. Ces derniers permettent le contrôle de l'habitat à distance et remontent l'information au propriétaire quand un événement pertinent survient.

1.4.4 Applications environnementales

La surveillance de l'environnement est l'une des applications les plus importantes des RCSFs, qui nécessite généralement une durée de vie de plusieurs mois, voire des années. Cependant, la restriction inhérente de l'énergie incluse dans la batterie des capteurs apporte une extrême difficulté pour obtenir une durée de vie de réseau satisfaisante, qui pourra devenir un goulot d'étranglement pour ce type d'applications.

Les récents changements climatiques et les catastrophes naturelles dans le monde ont suggéré l'importance de la surveillance de l'environnement, qui se développe rapidement comme une application majeure des RCSFs [82,83]. Par exemple, dans RCSFs, les capteurs peuvent être déployés dans un environnement hostile pour mesurer périodiquement les

paramètres météorologiques et hydrologiques autour de son environnement, tels que la lumière, la température, l'humidité, la vitesse et la direction du vent.

Bien que les RCSFs offrent plusieurs avantages dans le domaine de la surveillance de l'environnement, la conservation de l'énergie devrait toujours être prise en compte dans presque tous les domaines d'applications. La raison principale est qu'il est difficile voire impossible de recharger ou remplacer les batteries des capteurs dans un tel environnement. Par conséquent, l'économie d'énergie est devenue l'une des principales préoccupations dans la conception des RCSFs, et certains systèmes éco-énergétiques ont été développés pour atteindre l'objectif d'économie d'énergie.

Par ailleurs, une variété d'applications a été développée pour assurer la surveillance de l'environnement. Par exemple, une application pour comprendre la reproduction et le comportement d'un type d'oiseaux menacé sur l'île Great Duck aux Etats Unis [84], le projet Zebranet pour étudier le comportement de certains animaux en voie de disparition (lions, zèbres, etc.) [85], une application pour le suivi et la surveillance des mines de charbon souterraines [86], une application pour la surveillance et la détection des feux de forêt [87], une application pour collecter la température de l'eau de mer [88], une application pour la protection des cultures contre les intrusions d'animaux dans les terres agricoles [89].

1.4.5 Applications médicales

Une caractéristique essentielle des réseaux de capteurs corporels sans fil (WBAN) est son aptitude à offrir des communications pour les procédures de santé, en particulier WBAN insérés dans le corps humain. Les WBAN sont composés de capteurs de petite taille, légers et peu coûteux, qui peuvent être placés sur le corps comme des petits revêtements intelligents, incorporés dans les vêtements (vêtements intelligents), apposés sous la membrane, ou profondément implantés dans le tissu corporel [90,91]. Ces applications peuvent aider le personnel médical et les médecins dans le diagnostic et examiner soigneusement et continuellement l'état de santé des patients. Les données associées aux patients provenant des WBAN sont également liées à une source de soins centralisée pour des enregistrements stables, sûrs et fiables. Les données sont accessibles aux médecins pour déterminer l'état de santé du patient et peuvent aider à informer les patients des actions nécessaires par alarme, "Short Message Service (SMS)" ou e-mails de notification [91]. Au cours des dernières années, le nombre d'applications dédiées à la santé a augmenté telles que les applications permettant l'affichage des impulsions à distance, les applications pour les moniteurs de mouvement, les applications permettant la surveillance des maladies comme la maladie d'Alzheimer et les crises cardiaques [92], les applications pour la détection des crises d'épilepsie [93], les applications pour la surveillance des personnes âgées à distance, et les applications pour mesurer et surveiller les paramètres physiologiques des patients [94].

1.4.6 Applications domotiques

Le développement technologique a permis l'embarquement des capteurs dans des appareils utilisés quotidiennement dans une maison [95] tels que, les respirateurs, les réfrigérateurs, les machines à laver, etc. Ces capteurs peuvent communiquer entre eux ou avec un autre

réseau via Internet pour permettre aux utilisateurs de surveiller ces appareils domestiques localement ou à distance.

L'ensemble des technologies qui permettent d'automatiser et de faciliter la gestion des immeubles avec les systèmes embarqués, est désigné par le terme "Smart Home" [96–98] et cela dans le but de faciliter le mode de vie des utilisateurs et le rendre plus écologique avec un confort très élevé.

Avec l'avènement de l'IoT (Internet des Objets), le nombre d'équipements dotés d'une connexion s'est augmenté et a ouvert les portes à d'autres applications domotiques.

1.5 Technologies de communication dans les RCSFs

La communication dans les RCSFs exige des technologies qui prennent en compte des spécificités qui répondent aux exigences de ces réseaux. Dans ce qui suit, nous présentons des exemples de technologies sans fil qui s'adaptent aux RCSFs:

1.5.1 Bluetooth/IEEE 802.15.4

Bluetooth [99,100] est une technique de communication radio de courte distance, généralement utilisée dans les téléphones mobiles et destinée pour simplifier les connexions entre les appareils électroniques. Le seul inconvénient de cette technologie est sa grande consommation d'énergie. En plus, cette technologie ne pourrait pas connecter un grand nombre d'équipements. D'où, Bluetooth ne pourrait pas être utilisée dans les RCSFs à grande échelle.

1.5.2 Zigbee

Zigbee [101] est une technologie de communication sans fil de courte portée destinée pour les applications dans les réseaux personnels sans fil, principalement adoptée dans l'implémentation des RCSFs. La technologie Zigbee est caractérisée par sa faible complexité, faible coût, faible consommation d'énergie et faible taux de transmission de données, et aussi une compatibilité avec des dispositifs fixes ou mobiles. La norme IEEE 802.15.4 possède la plus faible demande de consommation d'énergie. C'est la norme idéale pour les équipements qui ont des ressources limitées tels que les RCSFs.

1.5.3 Dash7-ISO/IEC 18000-7

DASH7 [102] est une autre technologie de communication qui utilise la norme ISO/IEC 18000-7. Elle est caractérisée par sa faible consommation d'énergie ce qui permet d'augmenter la durée de vie de la batterie. Cette durée peut aller jusqu'à plusieurs années. Elle est caractérisée aussi par sa grande portée comparativement aux autres technologies de communication citées avant.

1.6 Architecture protocolaire

L'architecture protocolaire peut être utilisée par la station de base ou bien par les capteurs. Elle est constituée de cinq couches: une couche d'application, une couche de transport,

une couche réseau, une couche de liaison de données et une couche physique, et de trois plans: un plan de gestion d'énergie, un plan de gestion de mobilité et un plan de gestion des tâches [103]. La Figure 1.11 illustre les différentes couches et les différents plans dans la pile protocolaire. Cette pile [31] combine l'énergie et le routage, associe les données avec les protocoles actifs au niveau de la couche réseau et communique via un support sans fil. Les plans de gestion d'énergie, de mobilité et des tâches contrôlent et gèrent la consommation d'énergie, les mouvements et la répartition des tâches d'un capteur. Ces plans aident les capteurs à collaborer et partager les ressources entre eux, et aussi à équilibrer la consommation d'énergie pour prolonger la durée de vie du réseau.

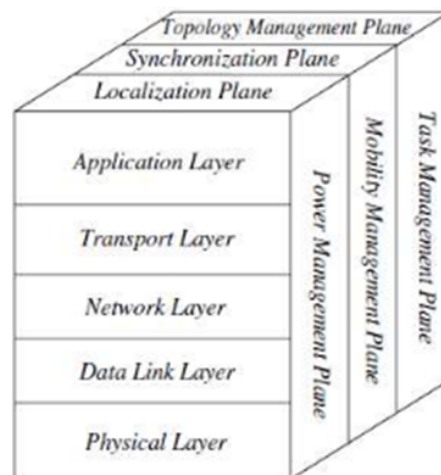


Figure 1.11: Pile protocolaire

La communication dans les réseaux de capteurs nécessite la mise en place d'un protocole de routage afin de gérer les transmissions de données effectuées par les capteurs pour atteindre la station de base. Cette gestion n'est pas facile car la topologie d'un réseau de capteurs est dynamique : un capteur pourrait devenir inaccessible quand il épuise sa batterie ou détériore ses capacités de communication à cause des conditions environnementales. Les protocoles de routage doivent utiliser des mécanismes permettant d'une part de minimiser la consommation d'énergie et d'autre part garantir la fidélité de routage c'est-à-dire entre tout capteur et la station de base il devrait exister au moins un chemin entre les deux noeuds.

1.7 Paradigme du routage dans les RCSFs

Dans les RCSFs, les capteurs sont déployés en grand nombre pour surveiller ou suivre des phénomènes (qui peuvent être soit des grandeurs physiques soit des mouvements ou autres) et faire remonter les informations de ces phénomènes à un centre de contrôle distant. Pour atteindre cet objectif, les capteurs ont la capacité de communiquer et de collaborer entre eux pour acheminer l'information collectée à la station de base. Cependant, le principal challenge dans les RCSFs est d'économiser le maximum d'énergie pour prolonger la durée de vie des réseaux en particulier si ces derniers sont déployés dans des zones hostiles avec absence de systèmes de récupération de l'énergie. Par ailleurs, sachant que l'opération de transmission de données par un capteur consomme plus d'énergie que

les autres opérations (traitement et réception). A cet effet, il est judicieux d'optimiser l'opération de routage de données dans les RCSFs pour conserver plus d'énergie et par conséquent augmenter leur longévité.

Le routage dans les RCSFs est différent par rapport au routage dans les réseaux traditionnels. Dans les RCSFs, il n'y a pas d'infrastructure, les liaisons ne sont pas fiables, les capteurs sont susceptibles de tomber en panne et les protocoles de routage doivent optimiser la consommation d'énergie et prolonger la durée de vie du réseau. Le rôle de chaque capteur est de collecter les données, les traiter et les transmettre à la station de base directement ou en utilisant une communication multi-sauts. Pour garantir la fiabilité de la communication avec une consommation d'énergie minimale un protocole de routage économe en énergie est nécessaire. Le routage optimal de l'information entre le nœud qui a collecté l'information et la station de base consiste à trouver les chemins les plus optimaux en termes de consommation d'énergie et de délai. Un chemin optimal, c'est le chemin qui contient le moins de sauts avec une consommation d'énergie minimale et un délai réduit.

Le type de collecte de données dans les RCSFs dépend de leur type d'applications. Certaines applications des RCSFs sont mises en place pour récolter des données de manière périodiques sur leur environnement alors que d'autres sont chargées de répondre à des événements importants produits à l'intérieur ou à proximité de la zone à surveiller, et d'autres encore sont conçues pour capter des informations précises sur des zones ciblées et répondre aux requêtes diffusées par les utilisateurs de l'application.

Par ailleurs, il existe des protocoles de routage qui se basent sur des algorithmes de recherche des plus courts chemins tels que Dijkstra et Bellman. Ces protocoles ont donné de meilleures performances lorsqu'il s'agit de réseaux de petite taille ou de taille moyenne. Cependant, lorsqu'il y a un passage à échelle (réseaux contenant un grand nombre de nœuds), les performances de ces algorithmes se dégradent. De ce fait, nous pouvons dire que les protocoles de routage qui se basent sur des algorithmes déterministes (traditionnels) ne sont pas efficaces pour les réseaux de grande dimension c'est la raison pour laquelle la communauté des chercheurs s'est orientée vers les algorithmes intelligents pour concevoir de nouveaux protocoles de routage pour les réseaux à grande échelle.

1.7.1 Les classes des protocoles de routage

Les protocoles de routage conçus pour les RCSFs peuvent être divisés en trois classes principales: les protocoles de routage à plat, les protocoles de routage hiérarchique et les protocoles de routage basés sur la localisation.

a) Routage à plat

Dans d'un protocole de routage à plat, tous les nœuds ont le même statut et le même rôle. La conception de ce type de protocoles est simple et facile. Néanmoins, elle présente quelques inconvénients tels que le manque de gestion des nœuds qui rend difficile l'optimisation des ressources du réseau et la lenteur de la réponse liées aux changements dynamiques du réseau. De ce fait, ce type de protocoles ne pourra pas convenir pour les réseaux à grande échelle.

b) Routage hiérarchique

Les protocoles de routage hiérarchique divisent les nœuds du réseau en plusieurs clusters. Chaque cluster est composé de plusieurs nœuds membres avec un seul clusterhead qui les supervise [104]. Les clusterheads sont sélectionnées parmi les nœuds de chaque cluster en fonction de certains critères. Les membres du cluster recueillent des informations et les transmettent à leur clusterhead. Les données sont ensuite agrégées au niveau des clusterheads ce qui évite des données redondantes. Chaque clusterhead transmet les données agrégées directement à la station de base, ou en utilisant d'autres clusterheads comme passerelles. Le clustering permet une utilisation efficace des ressources énergétiques et il est fortement recommandé pour les réseaux de grande dimension [105].

c) Routage basé sur la localisation

Dans le routage basé sur la localisation, les protocoles nécessitent des informations de localisation pour les nœuds et ces derniers sont adressés au moyen de leurs emplacements. Les informations de leurs emplacements sont nécessaires pour faciliter le calcul de la distance entre deux nœuds et pouvoir diffuser une requête vers une région particulière, éliminant ainsi le nombre de transmissions. Cela permet à son tour d'estimer la consommation d'énergie.

Dans les protocoles de routage basés sur la localisation, les nœuds doivent mémoriser les informations de localisation de leurs voisins. Par conséquent, ce processus peut entraîner une surcharge de routage (overhead) surtout s'il s'agit d'une topologie fortement dynamique.

1.7.2 Le routage dans les RCSFs à grande échelle

Le nombre de capteurs utilisés dans les RCSFs peut varier de quelques milliers à plusieurs dizaines de milliers. De plus, les RCSFs sont constitués sans aucune infrastructure prédéterminée. A cet effet, les schémas de routage dédiés aux RCSFs doivent être capables de fonctionner efficacement quel que soit la taille du réseau mis en place. Ces schémas doivent être capables de traiter un grand nombre d'envois de paquets sans être saturés. Un tel déploiement, nécessite que le protocole utilisé pour la communication soit capable de détecter les erreurs et de contrôler le flux [106]. Par ailleurs, lors de la transmission des données, les nœuds génèrent souvent des paquets supplémentaires à leurs propres paquets de données qui sont appelés paquets de contrôle. Dans un réseau classique, le nombre de paquets de contrôle est presque négligeable, mais il ne l'est pas de même pour un RCSF [107]. En effet, lorsque le nombre de nœuds augmente la surcharge du réseau devient de plus en plus importante ce qui signifie d'avantage de paquets de contrôle. D'où, un protocole de routage conçu pour les RCSFs à grande échelle doit pouvoir gérer cette surcharge, détecter les erreurs et les pannes et assurer la fiabilité des communications.

Lors du passage à l'échelle une communication multi-saut est suggérée puisque les capteurs sont limités en termes d'énergie et de portée de transmission. La quantité de l'énergie consommée lors d'une communication entre une paire de noeuds est influencée par la distance qui les sépare et le nature du milieu de propagation du signal. Par conséquent, la minimisation de la distance se réfère à la minimisation de la consommation d'énergie et à la prolongation de la durée de vie du réseau. De nombreux protocoles de routage ont

été proposés dans la littérature afin d'assurer des communications fiables et économes en énergie.

1.7.3 Apport des protocoles de routage intelligent

Le routage consiste à trouver un chemin optimal pour transmettre les données de la source à une destination. En utilisant une communication multi-saut, un nœud devrait être capable d'identifier le meilleur voisin (prochain saut) pour relayer le paquet afin qu'il atteigne la destination dans un délai très court et avec une consommation d'énergie minimale. La façon la plus simple d'une communication multi-saut serait d'inonder l'ensemble du réseau, mais il s'agit d'une approche irréalisable en raison de l'énergie limitée des capteurs et l'overhead causé par l'inondation. Il en résulte qu'un routage efficace fait face à de grands défis et qu'un protocole de routage doit tenir compte de certaines caractéristiques telles que:

- **Flexibilité:** elle définit la capacité d'identifier de nouveaux chemins de routage dans le cas où les nœuds des chemins existants soient déconnectés suite à des facteurs externes tels que la présence des obstacles temporaires ou permanents.
- **Passage à l'échelle:** il se définit par la capacité d'augmenter la taille du réseau sans influencer les performances du protocole.
- **Léger :** car les capteurs sont limités en termes de capacité de traitement.
- **Efficacité énergétique:** les nœuds fonctionnent avec des batteries limitées. De plus, les nœuds peuvent être déployés dans des zones hostiles où il est impossible de recharger leurs batteries.
- **Sans boucle:** l'algorithme de routage doit garantir l'arrivée des paquets envoyés à leur destination et assurer que les paquets ne soient pas acheminés en boucle (indéfiniment sur le réseau) sans trouver la bonne destination.
- **Fiabilité:** le protocole doit assurer un pourcentage élevé de transmissions réussies et garantir un taux de perte de paquets minimales.
- **Latence:** Certaines applications doivent recevoir les paquets demandés dans un certain délai. Cependant, il serait idéal d'avoir un protocole capable de trouver un compromis entre l'énergie consommée et le délai selon les exigences de l'application.

Toutes les caractéristiques ci-dessus impliquent la nécessité d'un protocole de routage intelligent pour les RCSFs.

1.8 Clustering dans les RCSFs

Le regroupement des capteurs en clusters a été largement adopté par la communauté des chercheurs pour satisfaire l'objectif passage à l'échelle et atteindre généralement une efficacité énergétique élevée et prolonger la durée de vie du réseau dans les environnements à grande échelle. Les protocoles de routage hiérarchique correspondants impliquent une

organisation en clusters des capteurs afin que l'agrégation des données soient possibles, conduisant ainsi à une importante économie d'énergie. Dans une architecture clusterisée, chaque cluster a un chef, également appelé CH et effectue généralement des tâches spéciales telles que la coordination entre les membres et l'agrégation des données, et plusieurs nœuds communs (SN) en tant que membres.

Le processus de formation des clusters conduit finalement à une hiérarchie à deux niveaux où les nœuds CHs forment le niveau supérieur et les nœuds membres des clusters forment le niveau inférieur. Les capteurs transmettent périodiquement les données à leurs CHs correspondants et ces derniers agrègent les données (diminuant ainsi le nombre total de paquets relayés) et les transmettent à la station de base (BS) soit directement, soit par une communication multi-sauts où d'autres CHs sont impliqués comme nœuds intermédiaires pour relayer les données envoyées. Cependant, comme les CHs envoient toutes les données temporelles à des distances plus élevées que les membres, ils dépensent alors plus d'énergie. Une solution courante pour équilibrer la consommation d'énergie entre tous les nœuds du réseau est de réélire périodiquement de nouveaux CHs (faisant ainsi tourner le rôle CH parmi tous les nœuds dans le temps) dans chaque cluster. Dans une architecture clusterisée, nous distinguons deux types de communications: communication intra-cluster et communication inter-cluster.

1.8.1 Principaux objectifs du clustering dans les RCSFs

Comme cela a été mentionné au début de cette section, le clustering dans les RCSFs peut grandement contribuer au passage à l'échelle, à la prolongation de la durée de vie et à l'efficacité énergétique de ces réseaux. Ainsi, le routage hiérarchique est un moyen efficace pour réduire la consommation d'énergie au sein d'un cluster, en effectuant une agrégation des données afin de réduire le nombre de messages redondants transmis à la station de base. Contrairement à un routage à plat qui peut entraîner une surcharge importante (overhead) avec l'augmentation de la densité des capteurs. Une telle surcharge peut entraîner une latence dans la communication et un suivi inadéquat des événements. De plus, une architecture basée sur un routage à plat ne permet pas le passage à l'échelle. Le clustering est particulièrement efficace pour les applications qui nécessitent un passage à l'échelle (des centaines ou des milliers de capteurs). Le passage à l'échelle dans ce contexte implique la nécessité d'un équilibrage de charge et d'une utilisation efficace des ressources.

En plus de prendre en charge le passage à l'échelle du réseau et de réduire la consommation d'énergie grâce à l'agrégation de données, le clustering présente de nombreux autres avantages [108]. Il peut localiser la configuration des routes au sein du cluster et ainsi réduire la taille de la table de routage stockée dans les nœuds. Il peut également conserver la bande passante de communication car il procure les interactions intercluster aux CHs et évite les échanges redondants de messages entre les capteurs. De plus, le clustering peut stabiliser la topologie du réseau au niveau des capteurs et ainsi réduire la surcharge de maintenance de la topologie. Les capteurs ne se soucieraient que de la connexion avec leurs CHs et ne seraient pas affectés par les changements au niveau des interactions entre CHs. Le CH peut également mettre en œuvre des stratégies de gestion optimisées pour améliorer encore le fonctionnement du réseau et prolonger la durée de vie de la batterie des capteurs et par conséquent la durée de vie du réseau. Un CH peut planifier des activités dans le cluster afin que ses membres associés puissent passer en

mode veille (basse consommation d'énergie) et réduire le taux de consommation d'énergie. En outre, les capteurs peuvent être engagés dans un ordre alternatif et le temps de leur transmission et de leur réception peut être déterminé comme dans LEACH, ainsi que la redondance de couverture peut être limitée et la collision d'accès au médium est évitée.

1.8.2 Propriétés du clustering

Dans cette sous-section nous présentons certaines propriétés du clustering. Ces propriétés servent également de moyen de base pour une comparaison et une catégorisation plus poussées des protocoles basés sur le clustering présentés dans la littérature.

- **Nombre de clusters:** Dans les algorithmes de clustering probabilistes les plus récents, le processus d'élection des CHs et de formation des clusters conduit naturellement à un nombre variable de clusters. Dans certaines approches l'ensemble des CHs est prédéterminé et donc le nombre de clusters est prédéfini. Le nombre de clusters est généralement un paramètre critique en ce qui concerne l'efficacité de l'ensemble des protocoles de routage. Il existe plusieurs approches qui ont été développées pour déterminer le nombre de clusters adéquat.
- **Communication intracluster:** Dans certaines approches de clustering, la communication entre un capteur membre et son CH associé est supposée être directe (communication à un saut). Cependant, une communication intracluster à sauts multiples est souvent requise lorsque la plage de communication des capteurs est limitée ou le nombre de capteurs est très grand et le nombre de CHs est petit.
- **Nœuds et mobilité des CHs:** si les membres et les CHs sont stationnaires, les clusters seront stables avec une gestion simplifiée pour les communications intracluster et intercluster. Au contraire, si les CHs ou les membres sont supposés mobiles, l'appartenance aux clusters pour chaque nœud devrait changer dynamiquement, forçant ainsi les clusters à évoluer dans le temps et doivent probablement être maintenus en permanence.
- **Types et rôles des nœuds:** dans les modèles de réseaux hétérogènes, les CHs sont supposés être équipés de beaucoup plus de ressources d'énergie, de calcul et de communication que les autres nœuds alors que dans les modèles de réseaux homogènes, tous les nœuds ont les mêmes capacités et seul un sous-ensemble des capteurs déployés est désigné comme CHs.
- **Stratégie de formation des clusters:** dans les approches les plus récentes, lorsque les CHs ne sont que des capteurs réguliers et que l'efficacité en temps est un critère de conception principal, la mise en clusters est réalisée de manière distribuée sans coordination. Dans quelques stratégies antérieures, une approche centralisée (ou hybride) était suivie; un ou plusieurs nœuds coordinateurs sont utilisés pour partitionner l'ensemble du réseau et contrôler l'appartenance des nœuds aux clusters.
- **Sélection des CHs:** les CHs dans certains algorithmes proposés principalement pour les environnements hétérogènes peuvent être préaffectés. Cependant, dans la plupart des cas où les environnements sont homogènes, les CHs sont choisis parmi

l'ensemble des nœuds déployés soit aléatoirement soit sur la base d'autres critères plus spécifiques tels que l'énergie résiduelle des capteurs, la connectivité, etc..

- **Plusieurs niveaux:** le concept d'une hiérarchie de clusters à plusieurs niveaux est introduit pour obtenir une distribution d'énergie et une consommation totale d'énergie encore meilleures au lieu d'utiliser un seul niveau de clustering. Les améliorations offertes par le clustering à plusieurs niveaux doivent être étudiées, en particulier lorsque nous avons de très grands réseaux et que l'efficacité de la communication inter-CH est très importante.
- **Chevauchement des clusters:** plusieurs protocoles accordent également une grande importance au concept de chevauchement de nœuds au sein de différents clusters soit pour une meilleure efficacité de routage, soit pour une exécution plus rapide du protocole de formation des clusters. Cependant, la plupart des protocoles connus essaient toujours d'avoir un chevauchement minimum entre les clusters.

1.9 Conclusion

Dans ce chapitre, nous avons présenté les réseaux de capteurs sans fil, quelques domaines d'applications de ce type de réseaux, leurs caractéristiques ainsi que les défis liés à leur conception.

Nous avons également fait un survol sur le routage dans les réseaux de capteurs à grande échelle qui fait l'objet de cette thèse. Puis, nous avons mis l'accent sur le concept de clustering qui constitue un avantage considérable pour les RCSFs en termes de réduction de consommation d'énergie d'une part, et d'autre part, il facilite le passage à l'échelle.

Dans le chapitre suivant, un état de l'art sur les solutions de routage proposées dans la littérature sera présenté. Cette étude nous permettra de proposer de nouvelles solutions de routage pour les RCSFs à grande échelle qui seront présentées dans les prochains chapitres.

Chapitre 2

État de l'art sur les protocoles de
routage dans les réseaux de capteurs

Chapter 2

État de l'art sur les protocoles de routage dans les réseaux de capteurs

2.1 Introduction

Un protocole de routage vise à trouver la meilleure méthode ou le meilleur algorithme pour acheminer les données d'une source à une destination. Par ailleurs, les critères de performances d'un RCSF sont en relation étroite avec le protocole de routage mis en oeuvre. Comme la durabilité des capteurs repose sur la durée de vie de leurs batteries et que ce type de réseaux est généralement déployé dans des endroits hostiles non accessibles par l'homme, le remplacement des batteries est difficile voire impossible. En outre, l'efficacité d'un protocole de routage conçu pour les RCSFs dépend de plusieurs critères tels que le passage à l'échelle, la tolérance aux pannes, la latence, la sécurité, la minimisation du taux de perte de données, la qualité de service. Néanmoins, l'efficacité énergétique reste l'enjeu majeur dans ce type de réseaux. De ce fait, l'utilisation des méthodes d'optimisation devient nécessaire car la réalisation d'un protocole de routage efficace en énergie peut être vu comme un problème NP-difficile dans les réseaux à grande échelle.

Dans ce chapitre, nous présentons un état de l'art sur les protocoles de routage dans les réseaux de capteurs sans fil avec des explications sur les méthodes utilisées pour leurs conceptions.

2.2 Taxonomie des protocoles de routage dans les RCSFs

Le routage consiste à envoyer des données d'une source à une destination en utilisant soit le mode de transmission mono-saut ou multi-sauts. La manière de concevoir un protocole de routage affecte fortement l'efficacité du réseau. Plusieurs protocoles de routage ont été proposés dans la littérature pour les RCSFs. A cet effet, il est possible de voir ces protocoles selon plusieurs classifications.

Bien qu'il existe de nombreux protocoles de routage pour les RCSFs, nous pouvons classer ces protocoles en deux grandes familles: les protocoles de routage basiques et les protocoles basés sur les méthodes d'optimisation. La première famille concerne les protocoles qui n'utilisent aucune méthode d'optimisation et ils sont généralement conçus

pour des réseaux de petite taille ou de taille moyenne alors que la deuxième famille inclue les protocoles qui impliquent des méthodes d'optimisation pour trouver le meilleur chemin entre la source et la destination parmi les chemins explorés. La Figure 2.1 présente une classification des protocoles de routage dans les RCSFs.

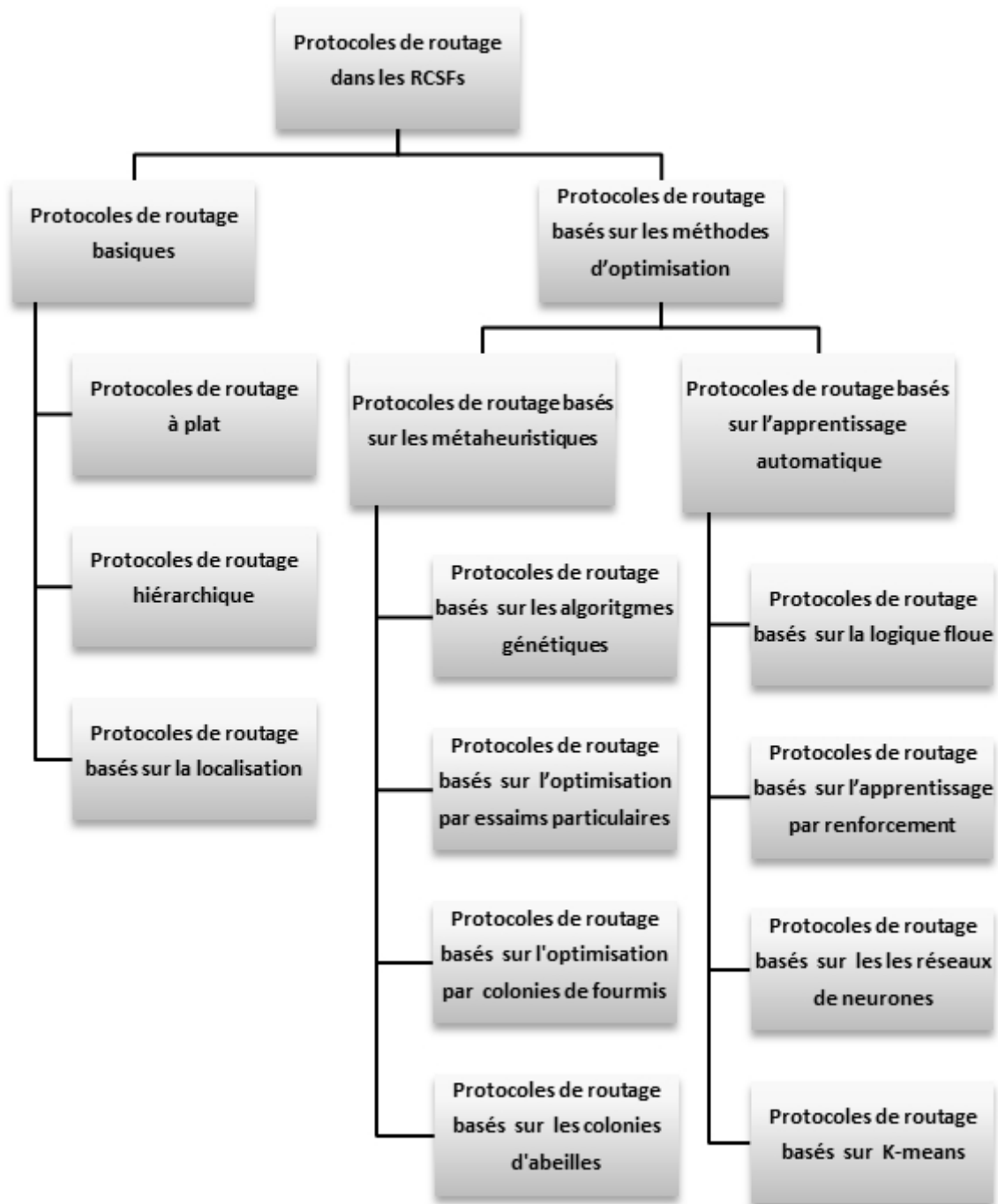


Figure 2.1: Classification des protocoles de routage dans les RCSFs

2.2.1 Les protocoles de routage basiques

Dans cette catégorie, nous présentons les protocoles de routage les plus populaires dans les RCSFs, qui ne sont pas basés sur des méthodes d'optimisation. Dans cette optique, trois sous-classes peuvent être définies en se basant sur la structure des réseaux qui sont les protocoles de routage à plat, les protocoles de routage hiérarchiques et les protocoles de routage basés sur la localisation.

a) Les protocoles de routage à plat

Dans ces protocoles de routage, tous les nœuds dans le réseau ont le même rôle et les mêmes ressources. Lorsqu'un nœud veut transmettre ses données à la station de base, il doit lancer une opération de recherche de chemins pour trouver un chemin valide, puis il transmet ses données en utilisant le mode de transmission multi-sauts. La mise en place de ce type de protocoles est simple comparativement aux autres types mais en contre partie il y'a une grande consommation d'énergie et un grand nombre de messages de contrôle (overhead) [109]. Dans ce qui suit, nous discutons certains protocoles de routage à plat.

1. Inondation et gossiping

Ces deux approches [110] sont des approches classiques pour la diffusion des données dans les RCSFs. Pour l'inondation, chaque nœud diffuse les paquets reçus pour la première fois de son nœud voisin. Il continue ce processus de diffusion de paquets jusqu'à ce que les paquets arrivent à destination ou jusqu'à ce que les paquets atteignent la condition de transmission (un nombre de sauts maximal). Pour l'approche basée sur le gossiping, le scénario change légèrement de celui de l'inondation où les données sont diffusées à des nœuds voisins sélectionnés aléatoirement et un autre voisin sélectionné au hasard envoie ces données, etc.

L'approche basée sur l'inondation présente certains inconvénients tels que l'implosion, le chevauchement et l'ignorance des ressources.

- L'implosion: aura lieu quand la réception d'un même message est dupliquée inutilement.
- Le chevauchement: est lié au déploiement dense des capteurs. En utilisant l'inondation, les capteurs se trouvant dans une même zone émettront tous la même donnée (ou presque) à la station de base.
- L'ignorance des ressources: cette approche ne prend pas en considération les ressources des nœuds lors de l'opération de diffusion.

2. Le protocole SPIN

"Sensor Protocol for Information via Negotiation (SPIN)" [111, 112] fait partie des premiers protocoles de routage centré-données (data-centric). L'idée derrière SPIN est de nommer les données à l'aide des descripteurs de haut niveau ou de métadonnées. Avant la transmission, les métadonnées sont échangées entre les capteurs via un mécanisme de publication de données, qui est la caractéristique clé de SPIN. Chaque nœud lors de la réception de nouvelles données, les communique à ses voisins et à ses voisins intéressés, c'est-à-dire ceux qui ne disposent pas des données, récupèrent

les données en envoyant un message de demande de données. La négociation des métadonnées de SPIN résout les problèmes classiques des inondations tels que la redondance de données et le chevauchement des zones de détection, ce qui permet d'atteindre une grande efficacité énergétique.

Il n'y a pas de format de métadonnées standard et il est supposé que ce format soit spécifique à l'application. Trois messages sont définis dans SPIN pour échanger des données entre les nœuds: un message ADV pour permettre à un capteur de publier une métadonnée particulière, un message REQ pour demander les données spécifiques et un message DATA qui transporte les données réelles. La Figure 2.2 résume les étapes du protocole SPIN.

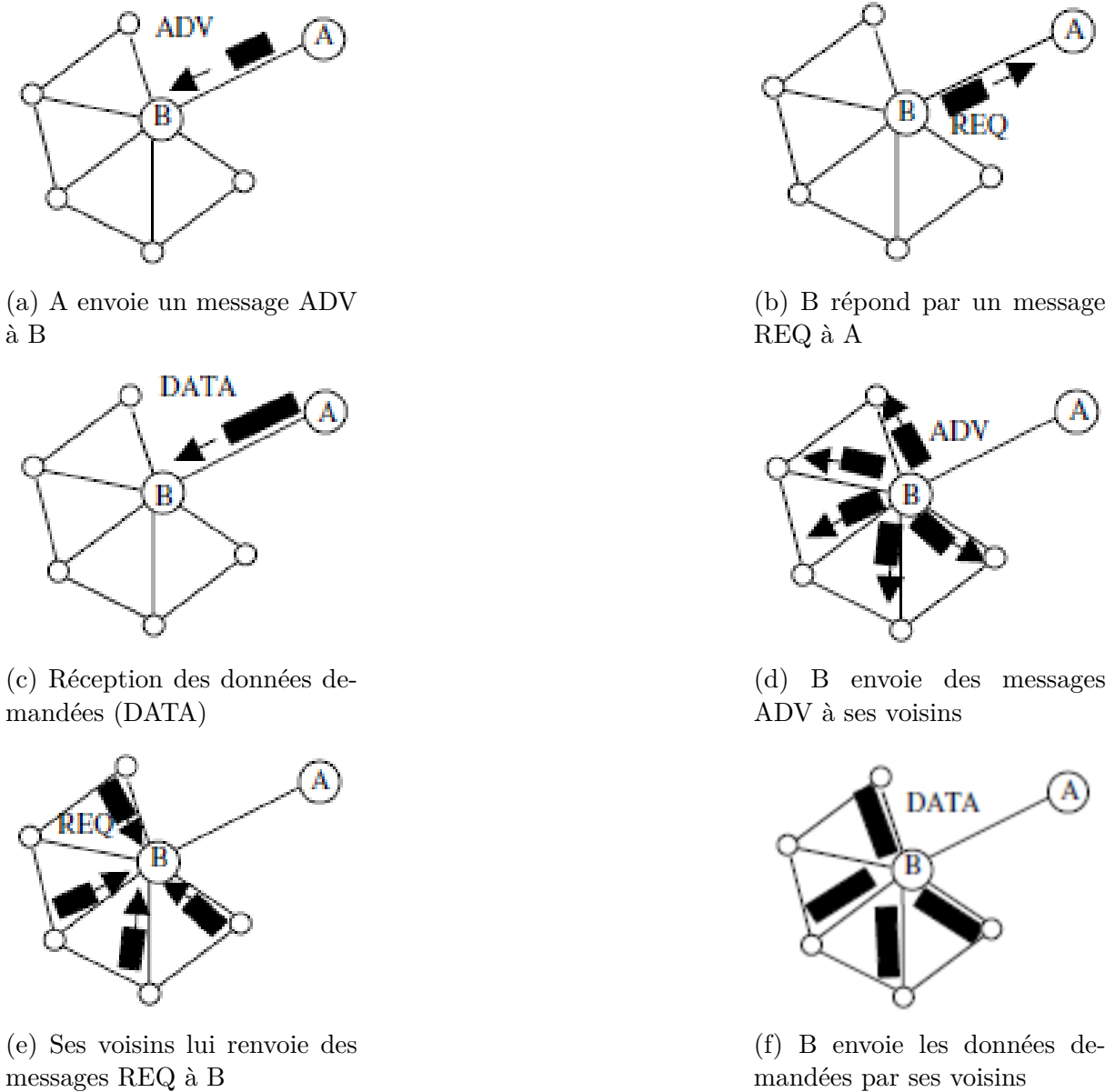


Figure 2.2: Le protocole SPIN

L'un des avantages de SPIN est que les changements de la topologie sont localisés car

chaque nœud n'a besoin de connaître que ses voisins directs (voisins à un saut). SPIN permet de minimiser de 3,5 fois la dissipation d'énergie et de la moitié les données redondantes comparativement à une inondation aveugle. Cependant, le mécanisme de publication des données de SPIN ne peut garantir la livraison des données. Par exemple, si les nœuds qui sont intéressés par les données sont loins du nœud source et que les nœuds entre la source et la destination ne sont pas intéressés par ces données, ces dernières ne seront pas du tout livrées à la destination. Par conséquent, SPIN n'est pas un bon choix pour des applications telles que la détection d'intrusion, qui nécessitent une livraison fiable de paquets de données à des intervalles de temps réguliers.

3. Le protocole "Directed Diffusion"

Le protocole "Directed Diffusion" [61] fait partie de la famille des protocoles de routage centré-donnée dans les RCSFs. Ce protocole vise à diffuser des données à travers des nœuds en utilisant un schéma de dénomination pour les données. La principale raison derrière l'utilisation d'un tel schéma est de se débarrasser des opérations inutiles de routage de la couche réseau afin d'économiser de l'énergie.

Le protocole "Directed Diffusion" suggère l'utilisation des paires "attribut-valeur" pour les données et interroge les capteurs à la demande en utilisant ces paires. Afin de créer une requête, un intérêt est défini à l'aide d'une liste de paires "attribut-valeur" telles que le nom des objets, l'intervalle, la durée, la zone géographique, etc. L'intérêt est diffusé par la station de base à travers ses voisins. Chaque nœud recevant l'intérêt peut effectuer la mise en cache (stocker) pour une utilisation ultérieure. Les nœuds ont également la capacité de faire une agrégation de données.

Les intérêts dans les caches sont ensuite utilisés pour comparer les données reçues avec les valeurs dans les intérêts. L'entrée d'intérêt contient également plusieurs champs de gradient où un gradient est un lien de réponse vers un voisin dont l'intérêt a été reçu. Il est caractérisé par le débit de données, la durée et le délai d'expiration dérivés des champs d'intérêt reçus. Par conséquent, en utilisant l'intérêt et les gradients, des chemins sont établis entre la station de base et les sources. Plusieurs chemins peuvent être établis pour que l'un d'eux soit sélectionné par renforcement. La station de base renvoie le message d'intérêt d'origine via le chemin sélectionné avec un intervalle plus petit, ce qui renforce le nœud source sur ce chemin pour envoyer des données plus fréquemment. Par ailleurs, la maintenance des chemins est possible dans le protocole "Directed diffusion". Lorsqu'un chemin entre un nœud source et la station de base échoue, un nouveau chemin ou un chemin alternatif doit être sélectionné.

Cependant, le protocole "Directed Diffusion" ne peut pas être appliqué à toutes les applications des réseaux de capteurs car il est basé sur un modèle de livraison de données orienté requêtes (query-driven). Les applications qui nécessitent une livraison continue de données à la station de base ne fonctionneront pas efficacement avec un modèle de données orienté requêtes. Par conséquent, ce protocole n'est pas un bon choix en tant que protocole de routage pour des applications telles que la surveillance environnementale. De plus, les schémas de dénomination utilisés dans ce protocole dépendent de l'application et doivent à chaque fois être définis à priori.

De plus, le processus de mapping entre les données et les requêtes peut nécessiter une surcharge supplémentaire au niveau des capteurs.

4. Le protocole SAR

"Sequential Assignment Routing (SAR)" [113] est l'un des premiers protocoles conçus pour les réseaux de capteurs sans fil qui a introduit la notion de "QoS" dans le processus de routage. SAR est basé sur l'association d'un niveau de priorité à chaque paquet. De plus, les liens et les chemins sont liés à une métrique qui caractérise leur fourniture potentielle de qualité de service. Cette métrique est basée sur le délai et le coût énergétique. Par ailleurs, SAR crée des arbres ayant comme racines les voisins à un saut de la station de base. Pour ce faire, plusieurs paramètres tels que la priorité des paquets, les ressources énergétiques et les métriques de QoS sont pris en compte. Le protocole SAR doit recalculer périodiquement les routes en cas de panne d'un des nœuds actifs. Cependant, l'inconvénient majeur de SAR est le problème de surcharge (overhead) car les tables de routage doivent être actualisées en continu pour faire face aux chemins non valides à un moment donné. Ce problème augmente à mesure que le nombre de nœuds augmente.

5. Le protocole CADR

"Constrained Anisotropic Diffusion Routing (CADR)" [114] est une version améliorée du protocole "Directed Diffusion" dans laquelle deux techniques ont été proposées: l'interrogation de capteurs basée sur l'information (IDSQ) et le routage contraint basé sur une diffusion anisotrope. CADR vise à acheminer plus de données vers la station de base tout en minimisant la latence et la bande passante. Ceci est réalisé en activant uniquement les capteurs qui sont proches d'un événement particulier et en ajustant dynamiquement les routes à travers lesquelles les données sont acheminées. La principale différence avec le protocole "Directed Diffusion" est la prise en compte du gain d'informations en plus du coût de communication. Dans CADR, chaque nœud évalue un objectif d'information/coût et achemine les données en fonction du gradient local associé à l'objectif "information/coût" et des exigences de l'utilisateur final. La mesure de l'utilité de l'information est modélisée à l'aide d'une estimation théorique.

IDSQ est basé sur un protocole dans lequel le nœud interrogateur peut déterminer quel nœud peut fournir les informations les plus utiles tout en équilibrant le coût énergétique. Bien que IDSQ offre un moyen pour sélectionner l'ordre optimal des capteurs pour permettre un gain d'informations, il ne définit pas spécifiquement comment la requête et les informations sont acheminées entre les capteurs et la station de base. Par conséquent, IDSQ peut être considéré comme une procédure d'optimisation complémentaire.

Par ailleurs, vu que CADR diffuse les requêtes en utilisant un ensemble de critères d'information pour sélectionner les capteurs pour obtenir les données, les résultats de simulations ont confirmé qu'il est plus économe en énergie que le protocole "Directed Diffusion" où les requêtes sont diffusées de manière isotrope, atteignant d'abord les voisins les plus proches.

6. Le protocole REEP

Le protocole "Reliable and Energy Efficient Protocol (REEP)" [115] permet de trans-

mettre les données via des chemins fiables et économes en énergie. Il se compose de cinq types d'événements: événement de détection, événement d'information, événement de requête, événement de la file d'attente des requêtes prioritaires et événement de valeur seuil de l'énergie. "L'événement de détection" est une demande produite par la station de base pour obtenir des informations. La réponse à cette demande est faite par "l'événement d'information" qui est produite par un capteur. Ensuite, les "événements de demande" sont utilisés pour définir les chemins. "l'événement de la file d'attente des requêtes prioritaires" est utilisé pour la sélection des chemins et "l'événement de valeur seuil de l'énergie" est utilisé pour vérifier les conditions des nœuds pour participer au processus de configuration des chemins.

Le protocole REEP est un protocole économe en énergie et permet une livraison fiable de données mais il génère une surcharge importante (overhead).

b) Les protocoles de routage hiérarchiques

L'objectif principal du routage dans les RCSFs est d'avoir un routage économe en énergie pour garantir une longue durée de vie pour ce type de réseaux. Les protocoles de routage hiérarchiques sont utilisés pour atteindre cet objectif. Ces protocoles sont définis comme des protocoles de routage basés sur le clustering, dans lesquels les nœuds jouent des rôles différents dans le réseau selon des critères bien définis. Généralement, le routage hiérarchique est une architecture de routage à deux niveaux où le premier niveau concerne le processus de clustering et le deuxième concerne le routage des données. Le clustering sert à diviser les RCSFs en clusters (groupes) en donnant à l'un des capteurs dans chaque cluster le rôle de "CH" et aux autres le statut de nœud membre. Dans chaque cluster, chaque nœud membre envoie les données captées à son CH correspondant. Un CH peut seulement relayer la donnée collectée par ses nœuds membres à la station de base comme il peut agréger l'ensemble des données reçues en un seul paquet et il le transmet à la station de base. Le clustering permet le passage à l'échelle et d'améliorer la durée de vie des réseaux. Dans ce qui suit, nous discutons certains protocoles de routage basés sur le clustering.

1. Le protocole LEACH

"Low Energy Adaptive Clustering Hierarchy (LEACH)" [24] est le protocole de routage le plus répandu parmi plusieurs protocoles de routage hiérarchiques (basés sur une architecture clusterisée) en termes d'efficacité énergétique. Dans cette sous-section, nous présentons brièvement ce protocole.

LEACH est un protocole de routage hiérarchique qui vise à minimiser la consommation d'énergie et ainsi à augmenter la durée de vie des réseaux de capteurs. LEACH fonctionne en plusieurs rounds où chaque round comprend une phase de configuration (set-up) et une phase de stabilité (steady-state). Pendant la phase de configuration, les nœuds sélectionnent un nombre aléatoire compris entre 0 et 1 et si ce nombre est inférieur à la valeur seuil $T(n)$, le nœud correspondant agira comme un CH pendant cette période (round). LEACH donne le rôle de CH en fonction de la valeur du seuil, calculée à l'aide de l'équation 2.1.

$$T(n) = \begin{cases} \frac{p}{1-p*(r \bmod \frac{1}{p})} & \text{if } i \in G \\ 0 & \text{otherwise} \end{cases} \quad (2.1)$$

Où p est le pourcentage des CHs, r est numéro du round courant, G est l'ensemble des nœuds qui n'ont pas été sélectionnés comme CH dans les derniers $(1/p)$ rounds.

LEACH s'exécute en deux phases: la phase de configuration et la phase de stabilité. Dans la phase de configuration, chaque CH diffuse un message pour construire son cluster et chaque nœud non-CH qui reçoit ce message rejoint le cluster adéquat en fonction de la puissance du signal "Received Signal Strength Indication (RSSI)" (indicateur d'intensité du signal reçu) du message reçu. Une fois les clusters formés, des time-slots selon une approche "Time Division Multiple Access (TDMA)" (accès multiple par répartition dans le temps) sont attribués aux nœuds membres de chaque cluster. Pour la deuxième phase, dans chaque cluster, chaque membre transmet ses données collectées à son CH correspondant durant son time-slot, et chaque CH agrège toutes les données reçues de ses membres dans un seul paquet de données et l'envoie directement à la station de base. Pour éviter les interférences entre les CHs, ces derniers communiquent avec la station de base selon la méthode d'accès "Code Division Multiple Access (CDMA)" (accès multiple de codage par division).

LEACH permet de réduire 7 fois plus la dissipation d'énergie par rapport à un schéma de routage basé sur une communication directe avec la station de base. LEACH est entièrement distribué et ne nécessite aucune connaissance globale de la topologie des réseaux. Cependant, LEACH utilise un routage à saut unique où chaque nœud peut transmettre directement les données à son CH correspondant et de même les CHs transmettent directement les données agrégées à la station de base. Par conséquent, il ne pourra pas être applicable aux réseaux déployés dans de grandes régions. De plus, le clustering dynamique utilisé dans LEACH entraîne une surcharge supplémentaire, par exemple le changement des CHs, le grand nombre de messages d'avertissement échangés, etc., peuvent diminuer le gain de consommation d'énergie.

2. Le protocole PEGASIS

"Power-Efficient Gathering in Sensor Information Systems (PEGASIS)" [7] est une amélioration du protocole LEACH. Plutôt que de former plusieurs clusters, PEGASIS forme des chaînes à partir des nœuds de sorte que chaque nœud transmet et reçoit d'un voisin et qu'un seul nœud est sélectionné dans cette chaîne pour transmettre à la station de base. Les données collectées se déplacent d'un nœud à l'autre, agrégées et éventuellement envoyées à la station de base. La construction de la chaîne est réalisée selon une approche gloutonne. Comme le montre la Figure 2.3, le nœud c_0 transmet ses données au nœud c_1 . Le nœud c_1 agrège les données du nœud c_0 avec les siennes, puis les transmet au leader. Une fois que le nœud c_2 a transmis le jeton au nœud c_4 , le nœud c_4 envoie ses données au nœud c_3 . Le nœud c_3 agrège les données du nœud c_4 avec les siennes, puis les transmet au leader. Le nœud c_2 attend de recevoir des données des deux voisins, puis agrège ses données

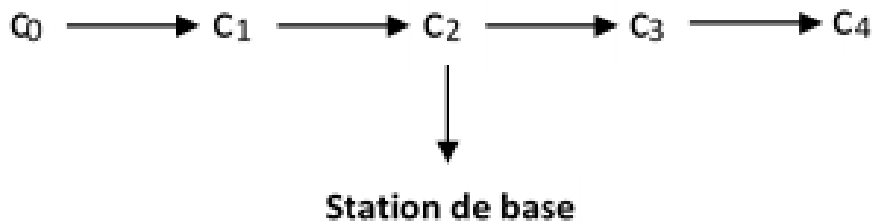


Figure 2.3: Chaînage dans PEGASIS [7]

avec les données de ses voisins. Enfin, le nœud c_2 transmet un message dans lequel toutes les données sont agrégées à la station de base.

La différence avec LEACH est d'utiliser un routage multi-sauts en formant des chaînes et en sélectionnant un seul nœud pour transmettre à la station de base au lieu d'utiliser plusieurs nœuds. Il a été démontré à travers des simulations que PEGASIS surpasse LEACH d'environ 100 à 300% pour différentes tailles et topologies de réseaux. Un tel gain de performances est obtenu grâce à l'élimination de la surcharge causée par la formation des clusters dynamiques dans LEACH et à la réduction du nombre de transmissions et de réceptions en utilisant l'agrégation de données. Cependant, PEGASIS introduit un délai excessif pour le nœud distant sur la chaîne. De plus, le leader unique peut devenir un goulot d'étranglement s'il tombe en panne.

3. Les protocoles TEEN et APTEEN

Le protocole "Threshold sensitive Energy Efficient sensor Network protocol (TEEN)" [8] est un protocole hiérarchique conçu pour répondre aux changements soudains des attributs détectés tels que la température et l'humidité. La réactivité est importante pour les applications temps réel, dans lesquelles le réseau fonctionne en mode réactif. TEEN se base sur une approche hiérarchique et utilise un modèle centré-données. L'architecture du réseau de capteurs est basée sur un regroupement hiérarchique où les nœuds les plus proches forment des clusters et ce processus passe au deuxième niveau jusqu'à ce que la station de base soit atteinte.

La Figure 2.4 représente le modèle de clustering utilisé par le protocole TEEN. Une fois les clusters formées, le CH diffuse deux seuils aux nœuds: "Hard" et "Soft". Le seuil "Hard" est la valeur minimale possible d'un attribut pour mettre en mode actif le transceiver d'un nœud et lui permettre de transmettre la donnée collectée à son CH correspondant. D'où, le seuil "Hard" permet aux nœuds de transmettre uniquement lorsque l'attribut détecté se trouve dans la plage d'intérêt, ce qui permet de réduire considérablement le nombre de transmissions. Une fois qu'un nœud détecte une valeur égale ou supérieure au seuil "Hard", il transmet des données uniquement lorsque la valeur de cet attribut change d'une valeur égale ou supérieure au seuil "Soft". En conséquence, le seuil "Soft" réduira encore le nombre de transmissions s'il y a peu ou pas de changement dans la valeur de l'attribut détecté. Par ailleurs, on peut ajuster à la fois les deux seuils afin de contrôler le nombre de transmissions de paquets. Cependant, le protocole TEEN n'est pas efficace pour les applications où des rapports périodiques sont nécessaires car l'utilisateur peut ne pas obtenir de données du tout si les seuils ne sont pas atteints.

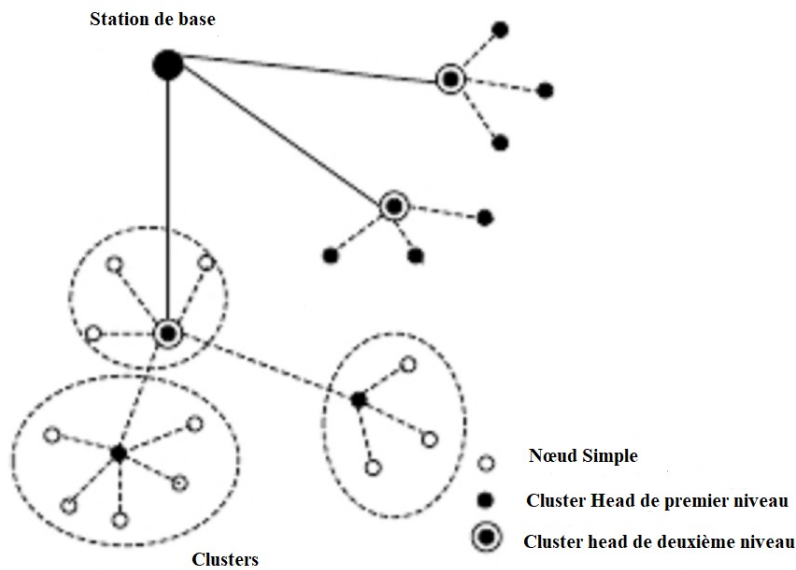


Figure 2.4: Clustering hiérarchique dans TEEN et APTEEN [8]

Le protocole "Adaptive Threshold sensitive Energy Efficient Sensor Network (APTEEN)" [116] est une extension de TEEN et vise à la fois à capturer des données périodiques et à réagir aux événements temporels. Son architecture est la même que celle du protocole TEEN comme montre la Figure 2.4. Lorsque la station de base forme les clusters, les CHs diffusent les attributs, les valeurs des seuils et l'échéancier de transmission à tous les nœuds. Les CHs effectuent également une agrégation de données afin d'économiser de l'énergie. Par ailleurs, le protocole APTEEN prend en charge trois types de requêtes: "historique", pour analyser les valeurs de données passées; "on-time", pour prendre une vue instantanée du réseau; et "persistant" pour surveiller un événement pendant un certain temps.

Les simulations ont démontré que les performances d'APTEEN sont meilleures comparées à LEACH et TEEN en termes de dissipation d'énergie et de durée de vie du réseau. Cependant, les principaux inconvénients des deux approches sont la surcharge (overhead) et la complexité de la formation des clusters à plusieurs niveaux et la mise en œuvre de fonctions basées sur des seuils.

4. Le protocole MLRC

Dans [117], les auteurs ont proposé un nouvel algorithme de clustering distribué multi-niveaux économe en énergie, appelé "An energy efficient Multi-Level Route-aware Clustering algorithm (MLRC)". Dans MLRC, pour établir un arbre parmi les nœuds, ce dernier applique une méthode soucieuse de chemins dans laquelle les nœuds peuvent obtenir les informations souhaitées sur les chemins possibles vers la destination. MLRC exécute trois phases: la phase de l'élection des CHs, la phase de la construction des routes (entre les CHs) et la phase pour la formation des clusters. Durant la première phase, tous les nœuds ayant un niveau d'énergie supérieur à l'énergie moyenne du réseau se concourent pour être des CHs. Les CHs sont choisis en fonction de leur énergie résiduelle et de leur distance vers la station de base

avec la méthode de normalisation min-max. Après l'élection des CHs, ces derniers forment entre eux des routes pour transmettre les paquets à la station de base selon un mode multi-sauts. Dans la dernière phase, chaque nœud non-CH rejoint un CH en fonction de la distance qui le sépare de ce dernier et le nombre de nœuds membres attachés à ce CH.

Le protocole MLRC élimine la génération supplémentaire des paquets de contrôle en implémentant la formation de clusters et la construction d'arbres de routage simultanément. Les CHs sont élus en fonction de paramètres efficaces: énergie résiduelle et la distance qui les sépare de la station de base. MLRC pourrait modérer la consommation d'énergie des relais proches de la station de base en attribuant une probabilité aux CHs adjacents et en évitant d'insister sur la sélection des CHs les plus proches. Les résultats expérimentaux ont montré que le protocole améliore la longévité des réseaux par rapport aux protocoles CEBCRA [118], EECF [119] et DHCR [120].

5. Le protocole HEEMP

"Hybrid Energy-Efficient Multi-path routing Protocol (HEEMP)" [121] est un protocole de routage qui exploite à la fois les avantages des schémas de routage à plat et hiérarchique pour maximiser l'efficacité énergétique dans les RCSFs. Dans HEEMP, la station de base sélectionne un certain nombre de nœuds souhaités comme CHs pour former des clusters dans le réseau. La sélection de ces CHs se fait en calculant une valeur CE (probabilité d'être CH) en fonction de l'énergie restante et la densité des nœuds. La station de base crée un ensemble de routes correspondant à chaque CH qui contient toutes les routes possibles entre ce CH et les autres CHs. Les routes sont établies qu'avec les nœuds qui ont un niveau d'énergie supérieur à un certain seuil. Une fois que l'établissement des chemins est achevé, le processus de routage des données pourra avoir lieu. À l'intérieur des clusters, les nœuds adoptent un schéma de routage multi-sauts pour communiquer avec leur CH correspondant qui, à la réception des paquets de données de tous les membres du cluster, transmet les données agrégées le long du chemin précalculé à la station de base. La communication intra-cluster peut se produire en deux modes: "philanthropist—maximal" où le nœud voisin ayant plus d'énergie résiduelle parmi les autres voisins est sélectionné comme prochain saut et "selfish—nearest" où le nœud le plus proche est sélectionné comme prochain saut. Par ailleurs, dans HEEMP, l'approche de routage évite aux nœuds de transmettre le long des liaisons longues, minimisant ainsi la consommation d'énergie du réseau.

Les résultats simulations ont montré que HEEMP surpasse le protocole M-TBC [122] en termes de longévité du premier nœud, d'énergie restante et de facteur de stabilité lors du passage à l'échelle.

6. Le protocole CAMP

Les auteurs de [123] ont proposé un protocole appelé "Cluster Aided Multi-Path routing protocol (CAMP)" dans lequel ils ont visé d'atteindre une déplétion énergétique uniforme sur tous les nœuds et donc de prolonger la durée de vie du réseau. CAMP divise la région d'intérêt en zones virtuelles et attribue un CH pour chacune. Dans l'ensemble du processus, un nœud peut soit faire partie d'un cluster, soit

rester une entité indépendante. Un non-membre du cluster transmet ses données au nœud suivant (prochain saut) à l'aide du processus de routage intelligent "Intelligent Routing Process (IRP)" (basé sur le compromis entre l'énergie résiduelle de lui-même et de son voisin, et l'énergie requise pour transmettre des paquets à son voisin). Si sur le chemin de transmission, un membre du cluster est élu en tant que prochain saut, il rejette le processus IRP et transmet les paquets à son CH correspondant, qui les transmet à son tour à la station de base (en adoptant une communication multi-sauts entre les CHs).

Les auteurs ont affirmé que le protocole CAMP proposé améliore la consommation d'énergie et la durée de vie comparé à d'autres protocoles de routage, à savoir, LEACH [24], PEGASIS [7], CEED [124] et CBMR [125] en raison de la sélection aléatoire des CHs ou en fonction de l'énergie résiduelle des nœuds. Cependant, avec de nombreux avantages, ce protocole présente un délai important en raison de son calcul de l'énergie et de la sélection aléatoire des CHs dans le réseau.

c) Routage basé sur la localisation

La plupart des protocoles de routage conçus pour les réseaux de capteurs nécessitent des informations de localisation pour les nœuds. Dans la plupart des cas, des informations de localisation sont nécessaires pour calculer la distance entre deux nœuds particuliers afin que la consommation d'énergie puisse être estimée. Comme il n'y a pas de schéma d'adressage pour les réseaux de capteurs comme les adresses "Internet Protocol (IP)" et qu'ils sont déployés spatialement dans une région, les informations de localisation peuvent être utilisées pour acheminer les données de manière économe en énergie. Par exemple, si la région à surveiller est connue, en utilisant l'emplacement des capteurs, la requête ne peut être diffusée qu'à cette région particulière, ce qui réduira considérablement le nombre de transmissions.

Dans ce type de protocoles, tous les capteurs calculent leurs distances avec leurs voisins en fonction de la puissance du signal (RSSI). Par ailleurs, il est possible d'obtenir les informations de localisation des capteurs à l'aide des systèmes de positionnement global "GPS". Chaque nœud qui veut envoyer un paquet, il ajoute la station de base dans son emplacement de destination. Les nœuds intermédiaires reçoivent ce paquet et l'envoient au prochain voisin à un saut qui est le plus proche géographiquement de la destination. Le processus se poursuit jusqu'à ce que les paquets de données arrivent à la station de base. Ce type de routage ne nécessite pas l'établissement ou la maintenance des chemins. Par conséquent, l'énergie et la bande passante sont conservées, car la demande de route et la propagation d'état ne sont pas nécessaires après une distance à un seul saut [126].

1. Le protocole GAF

"Geographic Adaptive Fidelity (GAF)" [9] est un protocole de routage basé sur la localisation sensible à l'énergie conçu principalement pour les réseaux mobiles ad hoc (MANETs), mais il peut également s'appliquer aux réseaux de capteurs. GAF conserve l'énergie en désactivant les nœuds inutiles du réseau sans affecter le niveau de fidélité de routage. Il forme une grille virtuelle pour la zone couverte. Chaque nœud utilise sa position indiquée par GPS pour s'associer à un point de la grille virtuelle. Les nœuds associés au même point sur la grille sont considérés comme équivalents en termes de coût de routage de paquets. Une telle équivalence est exploitée pour

garder certains nœuds situés dans une zone de grille particulière en état de sommeil afin d'économiser de l'énergie. Ainsi, GAF peut augmenter considérablement la durée de vie du réseau à mesure que le nombre de nœuds augmente. Un exemple de situation est représenté par la Figure 2.5 où le nœud 1 peut atteindre 2, 3 et 4 et les nœuds 2, 3 et 4 peuvent atteindre 5. Par conséquent, les nœuds 2, 3 et 4 sont équivalents et deux d'entre eux peuvent passer en mode veille.

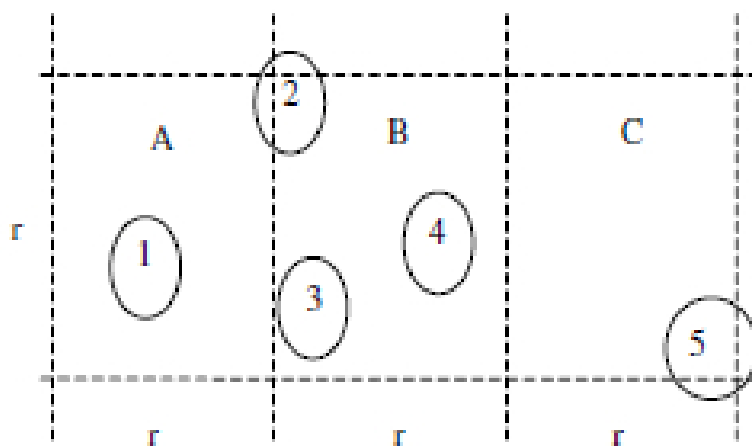


Figure 2.5: Exemple de grille virtuelle dans GAF [9]

Les nœuds changent d'état de veille à actif à tour de rôle afin que la consommation de l'énergie soit équilibrée entre eux. Trois états sont définis dans GAF: "*Découverte*", pour déterminer les voisins dans la grille, "*Actif*" reflétant la participation au routage et "*Veille*" lorsque la radio est éteinte. Le nœud qui passe en mode *Veille* et pour combien de temps dépendent de l'application et les paramètres associés sont ajustés en conséquence pendant le processus de routage. Afin de gérer la mobilité, chaque nœud dans une grille estime son temps de sortie du réseau et l'envoie à ses voisins. Les voisins endormis adaptent leur temps de sommeil en conséquence afin de conserver la fidélité de routage. Avant l'expiration du temps de départ du nœud actif, les nœuds endormis se réveillent et l'un d'eux devient actif. GAF est mis en œuvre à la fois pour la non-mobilité (GAF-basic) et la mobilité (GAF-mobilité adaptation) des nœuds.

GAF peut augmenter considérablement la durée de vie des réseaux. En effet, un seul nœud dans chaque grille reste à l'état actif en faisant passer les autres nœuds de la grille à l'état de sommeil pour une certaine période de temps tout en assurant la fidélité du routage. Cependant, dans certains environnements où les nœuds sont fortement mobiles, la fidélité du routage pourrait être réduite si un nœud actif quitte la grille. Ainsi, le nombre de données perdues sera important.

2. Le protocole GEAR

"Geographic and Energy Aware Routing (GEAR)" [127] utilise l'information géographique lors de la diffusion des requêtes aux régions cibles car les requêtes contiennent souvent des données géographiques. Son objectif est de réduire le nombre de

données dans le protocole "Directed Diffusion" en prenant en considération uniquement une certaine région, plutôt que d'envoyer les données à l'ensemble du réseau.

Avec le protocole GEAR, chaque nœud maintient un coût pour atteindre la destination en passant par ses voisins. Ce coût est divisé en deux parties : un coût estimé et un coût d'apprentissage. Le coût estimé est une combinaison de l'énergie résiduelle et de la distance jusqu'à la destination et le coût d'apprentissage est un raffinement du coût estimé qu'un nœud dépense pour le routage autour des trous dans le réseau. Un trou se forme quand un nœud n'a pas de voisin proche par lequel il peut atteindre la région cible. S'il n'y a pas de trous, le coût estimé est égal au coût d'apprentissage.

GEAR comporte deux phases:

- (a) Transfert de paquets vers la région cible: Lors de la réception d'un paquet, un nœud vérifie ses voisins pour voir s'il y a un voisin, qui est plus proche de la région cible que lui-même. S'il y en a plusieurs, le voisin le plus proche de la région cible est sélectionné comme prochain saut. S'ils sont tous plus éloignés que le nœud lui-même, cela signifie qu'il y a un trou. Dans ce cas, l'un des voisins est choisi pour transmettre le paquet en fonction du paramètre coût d'apprentissage. Ce choix peut ensuite être mis à jour en fonction de la convergence du coût appris lors de la livraison des paquets.
- (b) Transfert des paquets dans la région: Si le paquet a atteint la région, il peut être diffusé dans cette région soit par une transmission géographique récursive, soit par une inondation restreinte. Les inondations restreintes sont bonnes lorsque les capteurs ne sont pas déployés d'une manière dense. Dans les réseaux à haute densité, les inondations géographiques récursives sont plus économes en énergie que les inondations restreintes. Dans ce cas, la région est divisée en quatre sous-régions et quatre copies du paquet sont créées. Ce processus de fractionnement et de transmission se poursuit jusqu'à ce qu'il ne reste que les régions avec un seul nœud.

GEAR est comparé au protocole "Greedy Perimeter Stateless Routing (GPSR)" [128], qui est considéré comme l'un des premiers protocoles de routage géographique. GEAR offre de meilleures performances que GPSR en termes de livraison de paquets. Les résultats de simulations ont montré que pour une distribution de trafic inégale, GEAR délivre 70 à 80% de paquets de plus que GPSR alors que lorsque le trafic est uniforme, GEAR délivre 25 à 35% de paquets de plus que GPSR.

3. Le protocole LBRP

"Location-Based Routing Protocol (LBRP)" [129] utilise les informations de localisation et un schéma de routage basé sur un transfert glouton (greedy forwarding) pour acheminer les paquets de données d'un nœud source à un nœud destination. Dans ce protocole, certains nœuds enregistrent les emplacements de tous les nœuds du réseau. Ces nœuds sont appelés "Service de Localisation des Nœuds (SLN)". Lorsqu'un nœud veut envoyer un paquet de données, il déclenche d'abord le modèle "Module de Service de Localisation (MSL)" pour obtenir l'emplacement du nœud de destination et celui du prochain saut vers la destination à partir du SLN. Après

la réception d'informations par un nœud, ce dernier les envoie au prochain saut. Le même processus est répété jusqu'à ce que le paquet arrive à destination. Le schéma de routage "Greedy Forwarding" est utilisé pour sélectionner le nœud voisin le plus proche du nœud de destination et pour ainsi transmettre les paquets de données. LBRP améliore le débit du réseau mais il cause des surcharges dans la bande passante.

4. Le protocole LMOR

"Location-based Multi-objective Optimization Routing (LMOR)" [130] est un protocole qui vise à trouver un compromis entre les trois éléments de la qualité de service liés aux protocoles de routage basés sur la localisation. Ces trois éléments sont la consommation d'énergie, la qualité des liens et le passage à l'échelle. Une fonction d'optimisation multi-objectif est construite avec ces trois paramètres et le routage des paquets est basé sur son maximum. Les résultats simulations ont montré que le protocole LMOR réduit la consommation d'énergie moyenne de plus de 10% et prolonge la durée de vie du réseau de 45,2% par rapport au protocole de routage basé sur l'approche "Greedy Forwarding" traditionnel dans des réseaux où les liens ne sont pas complètement fiables.

2.2.2 Les protocoles de routage basés sur les méthodes d'optimisation

Dans cette section, nous présentons les protocoles de routage basés sur les méthodes d'optimisation en se focalisant plus sur les protocoles hiérarchiques avec une brève définition de chaque méthode d'optimisation.

Cette famille de protocoles se divise en deux classes qui sont les protocoles de routage basés sur les métaheuristiques et les protocoles de routage basés sur l'apprentissage automatique (Machine Learning).

a) Les protocoles de routage basés sur les métaheuristiques

Une métaheuristique est une méthode d'optimisation visant à résoudre des problèmes d'optimisation difficile là où le nombre de solutions réalisables ne peut pas être exploré dans un temps raisonnable. Grâce à cette dernière, il est possible d'avoir des solutions approchées pour des problèmes d'optimisation classiques de plus grande taille et pour de très nombreuses applications qu'il était impossible de les traiter auparavant [131].

Ces dernières années, l'intérêt porté aux métaheuristiques a pris de l'ampleur en recherche opérationnelle et en intelligence artificielle [132]. Bien qu'il existe plusieurs types de métaheuristiques, dans ce chapitre, nous nous intéressons aux principales métaheuristiques décrites dans la littérature.

Dans ce qui suit, nous présentons les principaux protocoles de routage basés sur les métaheuristiques qui sont conçus pour les RCSFs.

Les protocoles de routage basés sur les algorithmes génétiques

Les Algorithmes Génétiques (AGs) représentent une heuristique adaptative basée sur l'évolution génétique biologique pour une recherche et une optimisation intelligente. Les AGs modélisent l'évolution naturelle en effectuant des tests de condition physique sur de

nouvelles structures pour choisir la meilleure population [133]. Avec les AGs, une population est constituée par un groupe de chromosomes où chaque chromosome représente une solution complète à un problème donné, et une fonction fitness définit la qualité d'un chromosome en fonction de critères spécifiques. La première étape de ces algorithmes, est la génération de la population initiale qui peut être générée de manière aléatoire ou en utilisant une méthode avancée pour converger rapidement et obtenir une solution de meilleure qualité. L'objectif étant de parvenir à une solution proche de l'optimale, les AGs fonctionnent de manière itérative. A chaque itération, les chromosomes sont évalués en utilisant la fonction fitness selon des critères bien spécifiés et une nouvelle population est créée. La nouvelle population est générée en utilisant des méthodes de reproduction telles que le croisement et la mutation. Les solutions avec les valeurs fitness les plus élevées ont plus de chance d'être dans la nouvelle population. Dans le croisement, de nouveaux chromosomes sont créés en combinant entre les segments des chromosomes déjà existants (chromosomes parents). Dans la mutation, les chromosomes sont modifiés ou perturbés de manière aléatoire. À la fin de chaque itération, la nouvelle population obtenue est évaluée à l'aide de la fonction de fitness et les nouveaux chromosomes parents sont sélectionnés. Si la condition d'arrêt n'est pas satisfaite, une autre itération se déroule comme le montre la Figure 2.6.

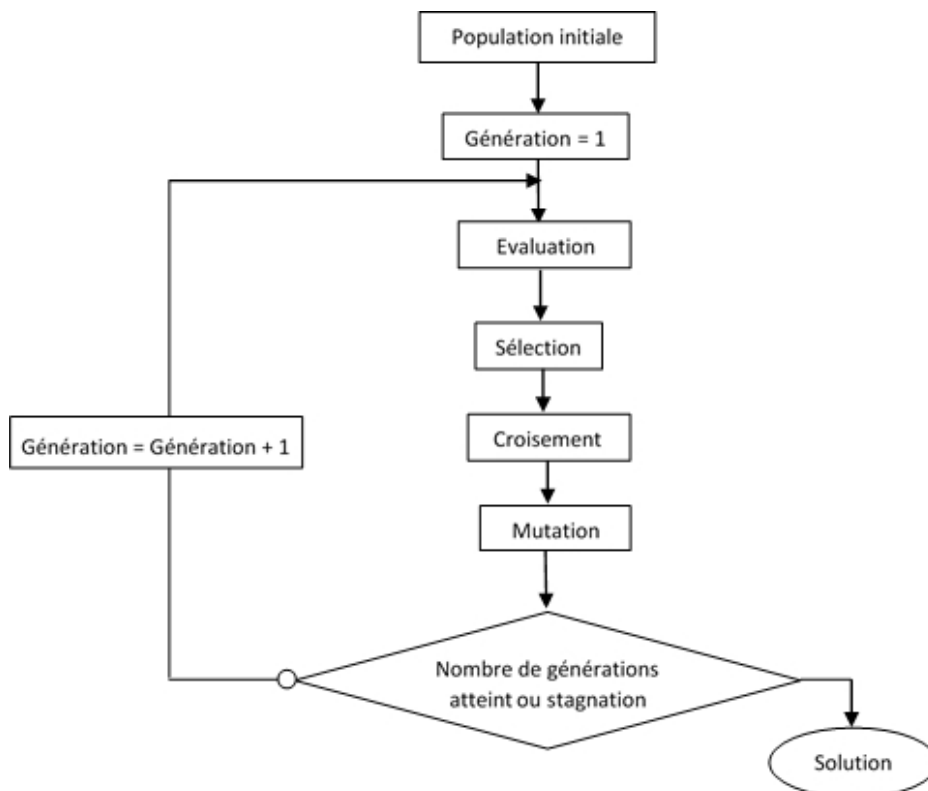


Figure 2.6: Organigramme des algorithmes génétiques

Ce type d'algorithmes est utilisé pour la recherche aléatoire et pour l'optimisation lors du routage des données. Les AGs ont fait preuve de souplesse dans la résolution des problèmes dynamiques. Ces algorithmes ont traité avec succès plusieurs problèmes NP-difficiles dans les RCSFs. Dans ce qui suit nous présentons quelques protocoles de routage conçus pour les RCSFs qui impliquent les AGs.

1. Le protocole GA-EECP

Dans "GA-based Energy-Efficient Clustering Protocol (GA-EECP)" [134], des techniques intelligentes ont été proposées pour la formation et la gestion des clusters. Les AGs sont utilisés pour déterminer le nombre de clusters, les CHs, les membres et l'ordonnancement des transmissions. La fonction objectif (fitness) utilisée dépend de la distance vers la station de base, de la distance vers le CH correspondant dans un cluster, de l'énergie de transmission et du nombre de transmissions. GA-EECP fournit de meilleures performances en termes de nombre de paquets transmis à la station de base et de durée de vie du réseau comparé aux protocoles LEACH [24], HCR-1 et HCR-2 [135]. Cependant, il a des exigences de traitement très élevées et il est difficile de définir une bonne évaluation de la fonction.

2. Le protocole LEACH-GA

"GA-Based Energy-Efficient Adaptive Clustering protocol (LEACH-GA)" [136] est une version améliorée de LEACH basée sur les AGs. En plus des deux phases exécutées (setup et steady-state) dans LEACH, LEACH-GA exécute une phase de préparation avant la première période (round). Pendant la phase de préparation, tous les nœuds exécutent initialement un processus de sélection des CHs, puis envoient leurs messages avec le statut d'être CH ou non, leurs identifiants (ID) ainsi que leurs positions géographiques par rapport à la station de base. Une fois la station de base reçoit les messages de tous les nœuds, cette dernière recherche ensuite une probabilité optimale que les nœuds soient des CHs via un algorithme génétique en minimisant la consommation d'énergie totale requise pour terminer la première période. Par la suite, la station de base diffuse un message d'avertissement avec la valeur optimale de probabilité à tous les nœuds afin de former des clusters dans la phase de configuration (setup) suivante. La phase de préparation n'est exécutée qu'une seule fois avant la phase setup de la première période.

Les résultats de simulations ont montré que le protocole LEACH-GA fournit efficacement une consommation d'énergie optimale pour les RCSFs, et permet une extension de la durée de vie du réseau comparativement à LEACH [24]. Cependant, la vitesse de convergence de ce protocole reste plus ou moins lente.

3. Le protocole GFCM

"Genetic algorithm Fuzzy C-Means (GFCM)" [137] combine l'algorithme génétique et l'algorithme "Fuzzy C-Mean (FCM)" pour diviser le réseau en clusters et pour l'élection des CHs. Un CH accepte uniquement les données en provenance des nœuds de son cluster. Après la réception de toutes les données, le CH fusionne ces données en un seul paquet et l'envoie à la station de base. Les clusters deviennent fixes après une période de temps. Dans GFCM, l'AG est utilisé pour définir les valeurs initiales de FCM. GFCM fournit de meilleures performances comparé à LEACH [24] en termes de consommation d'énergie et durée de vie du réseau.

4. Le protocole GABEEC

"GA-Based Energy Efficient Clusters (GABEEC)" [138] implique les AGs pour avoir des clusters efficaces en énergie dans les RCSFs afin d'améliorer la durée de vie dans ce type de réseaux. GABEEC s'exécute en deux phases : la phase set-up et la phase steady-state. Dans la première phase, les clusters sont créés une seule fois

durant la durée de vie du réseau, mais les CHs changent dynamiquement en fonction de l'énergie résiduelle. GABEEC utilise une représentation binaire du réseau où chaque bit représente le statut de chaque nœud: "1" si le nœud est un CH et "0" s'il est un nœud simple. Une instance du réseau représente un chromosome spécifique et l'AG évalue chaque chromosome et sélectionne le meilleur profil pour que l'énergie soit exploitée efficacement et la durée de vie du réseau soit prolongée. GABEEC est d'abord comparé à LEACH (avec 100 nœuds), puis au protocole HCR [139] et au protocole proposé dans [134] (avec 200 nœuds). Les résultats ont montré que GABEEC prolonge la durée de vie des réseaux mieux que ces protocoles dans les deux scénarios. Cependant, la manière de représenter un réseau par les algorithmes génétiques dans GABEEC reste plus ou moins non pratique pour les réseaux à grande échelle.

5. Le protocole M2NGA

"Multi-objective Two-Nested Genetic Algorithm (M2NGA)" [140] propose une méthode optimale de clustering dans les RCSFs homogènes en utilisant un algorithme génétique multi-objectif à deux niveaux. L'AG est utilisé pour trouver une topologie efficace dans chaque cluster et pour trouver les meilleurs chemins entre les clusters. Le premier niveau de l'algorithme est un algorithme génétique multi-objectif dont le but est d'obtenir des schémas de clustering dans lesquels la durée de vie du réseau et le délai sont optimisés. L'AG de bas niveau est utilisé dans chaque cluster afin d'obtenir la topologie la plus efficace pour la transmission des données des capteurs à leur CH correspondant. M2NGA consomme beaucoup moins d'énergie par rapport à LEACH [24] et à LA2D-GA [141]. Cependant, il n'était pas évalué dans des réseaux de grande taille.

6. Le protocole GAR

"An Energy Efficient GA-Based Routing protocol (GAR)" [142] est un protocole de routage conçu pour les RCSFs. GAR implique les AGs pour traiter le problème de routage des données avec l'agrégation des données dans les RCSFs. Ce protocole cherche à conserver l'énergie des nœuds relais en déterminant les chemins de tous les nœuds relais vers la station de base avec la minimisation de leurs distances globales. L'avantage de ce protocole est de calculer rapidement un nouveau schéma de routage en fonction de l'état du réseau. Les résultats de simulations ont montré que GAR surpasse MHRM [143] en termes de durée de vie du réseau et de consommation d'énergie. Cependant, le schéma de routage proposé par GAR, ne lui permet d'être un protocole tolérant aux pannes en particulier si un nœud relais tombe en panne.

7. Le protocole OZEED

Le protocole "Optimized Zone based Energy Efficient routing Protocol (OZEED)" [144] est une amélioration du protocole ZEEP [145]. OZEED vise à optimiser les processus de clustering et de sélection des CHs proposés dans ZEEP en utilisant un système génétique combiné avec la logique floue. Le processus de clustering est divisé en deux étapes. La première étape consiste en un système d'inférence floue pour sélectionner les nœuds optimaux qui peuvent être des CHs en fonction de certains paramètres tels que l'énergie, la distance, la densité et la mobilité. Dans la deuxième étape, un algorithme génétique est utilisé pour faire un choix final des

CHs parmi les candidats proposés par le système flou afin que la solution optimale générée soit un ensemble équilibré de clusters uniformément répartis permettant d'améliorer la durée de vie du réseau. Les résultats de simulations ont montré que OZEEP fournit de meilleures performances en termes de nombre de paquets perdus, consommation d'énergie et durée de vie du réseau par rapport au protocole ZEEP.

8. Le protocole ASLPR

"A new evolutionary based application specific routing protocol (ASLPR)" [146] est un protocole de routage économe en énergie conçu pour des applications spécifiques dans les RCSFs. ASLPR prend en compte certaines propriétés des nœuds telles que la distance qui les sépare de la station de base, l'énergie résiduelle, la distance entre les CHs pour élire les CHs d'une manière optimale. Il se base sur un algorithme hybride qui combine deux métaheuristiques : les AGs et le recuit simulé "Simulated Annealing (SA)". Cet algorithme est appliqué afin d'optimiser les paramètres contrôlables du protocole dans le but de choisir des CHs. Par ailleurs, puisque l'AG est un algorithme basé sur la population, il a une bonne exploration globale de l'espace de recherche. D'autre part, le recuit simulé a une très bonne exploration locale dans l'espace de recherche, grâce à son mécanisme de recherche locale et à son évolution à solution unique. De ce fait, dans un premier temps, l'AG est exécuté pour réaliser une recherche globale dans l'espace des solutions. Ensuite, le SA est appliqué pour raffiner localement le processus de recherche autour de la solution finale trouvée par l'AG. Les résultats de simulations ont montré l'efficacité de la méthodologie proposée dans ASLPR pour équilibrer la consommation d'énergie des nœuds et maximiser la durée de vie du réseau par rapport à LEACH [24], LEACH-DT [147] et LEACH-EP [148].

9. Le protocole de routage basé sur GA et K-Means

Dans [149], les auteurs ont proposé une approche de clustering qui combine les AGs améliorés et la méthode K-means pour réduire la consommation d'énergie et prolonger la durée de vie des réseaux de capteurs. L'approche proposée utilise une version améliorée des AGs pour trouver le nombre optimal de CHs. En outre, elle utilise la méthode K-means pour former des clusters tout en équilibrant la distribution d'énergie entre eux. Les simulations ont montré que ce protocole proposé permet une durée de vie de réseau plus longue que dans d'autres protocoles tels que LEACH [24], GAEEP et GABEEC [138].

10. Le protocole GADA-LEACH

"A Genetic Algorithm based Distance-Aware routing protocol (GADA-LEACH)" [150] est un protocole de routage qui implique un algorithme génétique pour améliorer la sélection des CHs dans le protocole LEACH. Pour ce faire, le concept de nœud relais a été introduit, qui sert d'intermédiaire entre le CH et la station de base pour faciliter la communication entre le CH et la station de base. Les résultats de simulations obtenus ont confirmé que GADA-LEACH est efficace en termes de durée de vie du réseau comparativement à LEACH [24], LEACH-E [151] et LEACH-EX [151]. Cependant, dans ce travail les auteurs n'ont pas prouvé que la sélection des CHs a été faite d'une manière optimale.

11. Le protocole basé sur une approche hybride (KGA)

Dans [152], une approche hybride combinant un algorithme génétique (GA) et la méthode K-Means, appelé KGA, a été proposée pour concevoir un protocole de routage avec une efficacité énergétique dans les RCSFs hétérogènes. Dans ce travail, les auteurs ont considéré que le problème du clustering optimal est en relation étroite avec le problème de recherche d'un nombre optimal de clusters dans un grand espace de recherche de sorte que les métriques de performances dans les RCSFs soient optimisées. Dans le protocole proposé, la distance entre les clusters, la distance à l'intérieur des clusters et un certain nombre de CHs sont utilisées pour rechercher le nombre optimal de clusters et de CHs. La maximisation de l'économie d'énergie et la durée de vie d'un réseau sont considérés comme étant les deux paramètres importants. En outre, le protocole KGA est conçu avec une approche hybride du schéma d'initialisation de la population et de la fonction objectif.

Dans KGA, la fonction objectif est définie en fonction des distances intra et inter cluster et du nombre de nœuds vivants dans le réseau selon l'équation 2.2

$$Min(f) = \alpha \left(\frac{D_{intra}}{D_{inter}} \right) + \beta |WSN_C| \quad (2.2)$$

où, α et β sont des coefficients pondérés tels que $\alpha + \beta = 1$. D_{intra} et D_{inter} sont respectivement les distances intra-cluster et inter-cluster données par les équations 2.3 et 2.4, et $|WSN_C|$ est le nombre de CHs dans le réseau.

$$D_{intra} = \sum_{i=1}^{CHs} \sum_{\forall S_j \in C_i} d(S_j, CH_i) \quad \text{where } j = 1, \dots, n \quad (2.3)$$

$$D_{inter} = \sum_{\forall C_i, C_j, C_i \neq C_j} d(CH_i, CH_j) \quad (2.4)$$

Les résultats de simulations ont montré des performances meilleures comparées à IHCR, SEP [153] en termes de durée de vie du réseau, d'énergie résiduelle et débit.

12. Le protocole O-LEACH

"Genetic algorithm based optimized leach protocol (O-LEACH)" [154] est une variante du protocole LEACH [24]. Dans ce protocole, la phase de la formation des clusters est similaire à celle de LEACH par contre la phase d'élection des CHs et la phase de transmission des données (steady-state phase) sont différentes. Les CHs sont élus en fonction de leur niveau de batterie. Dans la phase de transmission des données, chaque nœud membre d'un cluster envoie les informations collectées à son CH correspondant selon son échéance TDMA time-slot et chaque CH agrège les paquets en un seul paquet et le transmet à la station de base en utilisant un mode de transmission multi-sauts. Dans ce protocole, l'AG est utilisé pour trouver la route optimale inter-cluster (CH-to-CH) en utilisant une fonction fitness qui prend l'énergie résiduelle des CHs comme paramètre. Les résultats de simulations sous Matlab ont montré que le protocole O-LEACH réduit le taux de consommation d'énergie de 17,39% comparé à LEACH. Cependant, les auteurs ont utilisé les AGs

d'une manière simple ce qui n'a pas permis d'obtenir des résultats très satisfaisants par rapport à LEACH.

Les protocoles de routage basés sur l'optimisation par essaims particulaires

"Particle Swarm Optimization (PSO)" est une technique d'optimisation qui est inspirée d'un comportement social des animaux évoluant en essaims tels que les essaims d'abeilles, vols groupés d'oiseaux, bancs de poissons [155]. Les particules d'un même essaim communiquent entre elles tout au long de la recherche pour construire une solution au problème posé en s'appuyant sur leur expérience collective. PSO est simple à comprendre, à programmer et à utiliser, et se révèle particulièrement efficace pour les problèmes d'optimisation. Cependant, comme toutes les métaheuristiques, PSO possède des inconvénients tels que le problème de convergence prématurée, qui peut conduire les algorithmes de ce type à stagner dans un optimum local.

Comme les autres techniques évolutionnaires, PSO est une technique de recherche basée sur la population et est initialisée avec une population de solutions aléatoires, appelées particules. Une particule aura une valeur de condition physique, qui sera évaluée par une fonction objectif à optimiser pour chaque génération [156,157]. La Figure 2.7 illustre la chronologie d'exécution des différentes étapes dans PSO.

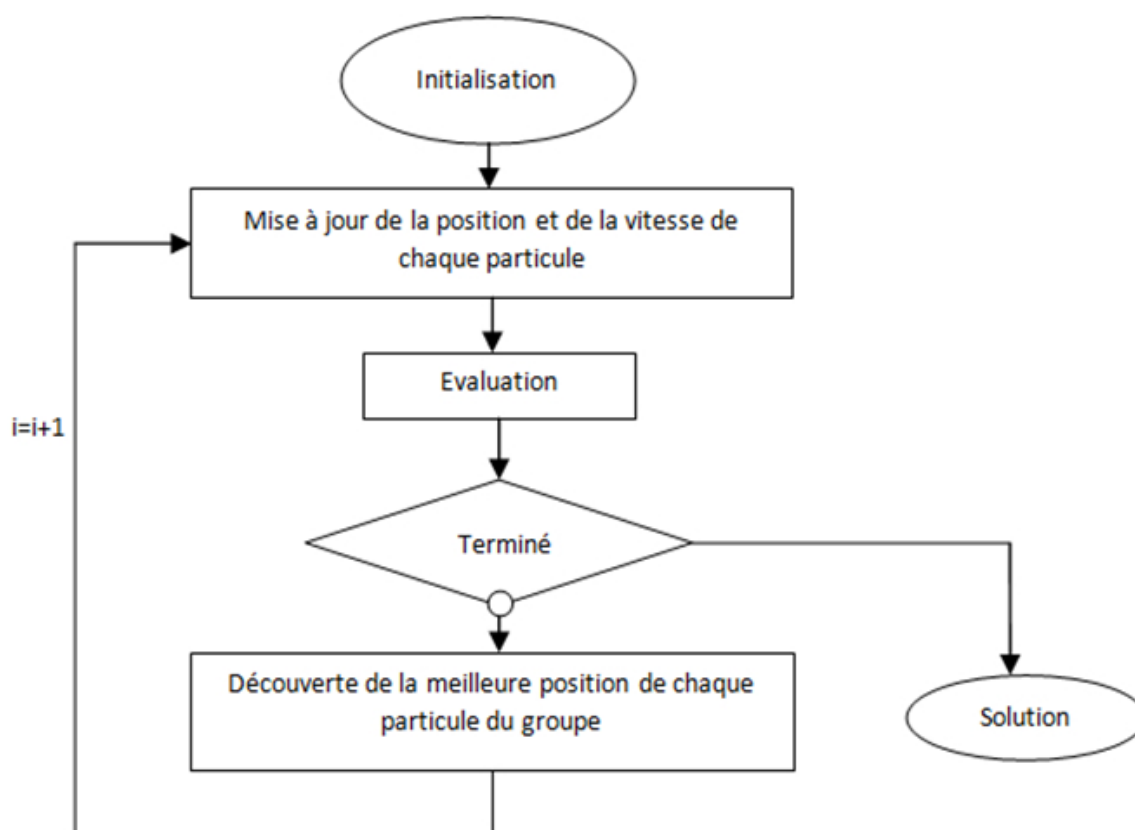


Figure 2.7: Organigramme de l'algorithme d'optimisation par PSO

Pour trouver la solution à un problème donné, PSO s'appuie sur l'expérience collective des différentes particules d'un essaim qui communiquent entre elles. Ainsi, afin

d'augmenter les performances des RCSFs, plusieurs protocoles de routage basés sur l'optimisation par essais particuliers ont été proposés dans la littérature. Dans ce qui suit, nous présentons et nous discutons quelques protocoles de routage conçus pour les RCSFs et qui impliquent la technique PSO.

1. Le protocole PSO-C

"Energy-aware clustering for WSNs using PSO (PSO-C)" [158] est un protocole de routage basé sur le clustering et implique la technique PSO pour augmenter les performances des RCSFs. Le fonctionnement de ce protocole est basé sur un algorithme de contrôle centralisé qui est implémenté au niveau la station de base. PSO-C est exécuté en plusieurs périodes, où chaque période (round) commence par une phase de configuration (set-up) au cours de laquelle des clusters sont formés. Cette phase est suivie d'une phase d'état stationnaire (steady-phase) dans laquelle une approche similaire à celle de [24] est utilisée. Au début de chaque phase de configuration, tous les nœuds envoient des informations sur leurs états énergétiques actuels et leurs emplacements à la station de base. Sur la base de ces informations, la station de base calcule le niveau d'énergie moyen de tous les nœuds. Ainsi, pour garantir que seuls les nœuds avec une énergie suffisante sont sélectionnés comme CHs, les nœuds avec un niveau d'énergie supérieur à la moyenne sont éligibles pour être des CHs candidats pour la période courante. Ensuite, la station de base exécute l'algorithme PSO pour déterminer les K meilleurs CHs qui peuvent minimiser la fonction coût définie par l'équation 2.5.

$$\text{coût} = \beta f_1 + (1 - \beta) f_2 \quad (2.5)$$

$$f_1 = \max_{k=1,2,\dots,K} \left\{ \sum_{\forall n_i \in C_{p,k}} \frac{d(n_i, CH_{p,k})}{|C_{p,k}|} \right\} \quad (2.6)$$

$$f_2 = \frac{\sum_{i=1}^N E(n_i)}{\sum_{k=1}^K E(C_{p,k})} \quad (2.7)$$

où f_1 est la distance Euclidienne moyenne maximale des nœuds membres à leurs CHs associés et $|C_{p,k}|$ est le nombre de nœuds qui appartiennent au cluster C_k de la particule p . La fonction f_2 est le rapport de l'énergie initiale totale de tous les nœuds $n_i, i = 1, 2, \dots, N$ dans le réseau avec l'énergie totale actuelle des CHs candidats dans la période en cours. β est une constante définie par l'utilisateur pour peser la contribution de chacun des sous-objectifs. La fonction fitness définie ci-dessus a pour objectif de minimiser simultanément la distance intra-cluster entre les nœuds membres et leurs CHs correspondants, telle que quantifiée par f_1 ; et aussi d'optimiser l'efficacité énergétique du réseau quantifiée par f_2 . Selon la fonction coût définie par l'équation 2.5, des petites valeurs de f_1 et f_2 suggèrent des clusters compacts avec un ensemble optimal de nœuds qui ont suffisamment d'énergie pour effectuer les tâches de CHs.

A travers des simulations, PSO-C a été évalué et comparé à LEACH [24] et à LEACH-C [159] sur deux scénarios différents. Dans le premier scénario, la station

de base est située à l'intérieur du champ de captage, tandis dans le second scénario, elle est située à l'extérieur. Les résultats de simulations ont montré que PSO-C améliore le nombre de paquets reçus par la station de base dans le premier scénario à 101 % et 25 % respectivement que LEACH et LEACH-C, et dans le deuxième scénario 60 % et 50 %. PSO-C améliore également la durée de vie du réseau en améliorant le nombre de nœuds vivants dans les deux scénarios. Cependant, cette proposition n'était pas évaluée dans réseaux de grande dimension pour voir si elle garde les mêmes performances lors du passage à l'échelle.

2. Le protocole PSO-HC

Dans [160], les auteurs ont proposé un nouveau protocole centralisé basé sur un algorithme PSO pour le clustering hiérarchique "PSO-Hierarchical Clustering (PSO-HC)" dans les RCSFs. L'objectif visé dans PSO-HC est de maximiser la durée de vie du réseau en minimisant le nombre de CHs actifs et de permettre le passage à l'échelle en utilisant une communication à deux sauts entre les nœuds membres et leurs CHs correspondants.

Dans PSO-HC, le temps de fonctionnement du réseau est divisé en périodes. Chaque période se compose de deux phases, la phase d'installation (set-up) et la phase de régime permanent (steady-state).

- *Phase set-up* : La phase set-up commence par la découverte des voisins où chaque nœud du réseau diffuse un paquet HELLO contenant son ID. Un nœud qui reçoit ce paquet HELLO mettra à jour sa table de voisinage avec l'ID inclus dans le paquet ainsi la valeur RSSI. Après que la découverte des voisins par tous les nœuds, le protocole utilise la méthode d'inondation pour transférer les données de contrôle à la station de base. Chaque nœud diffuse les données suivantes sur lui-même: ID, énergie résiduelle et la table voisinage. Un nœud qui reçoit ce paquet pour la première fois le rediffusera jusqu'à ce qu'il atteigne la station de base. Sur la base des informations reçues par la station de base, cette dernière calcule le niveau d'énergie moyen de tous les nœuds. Seuls les nœuds ayant un niveau d'énergie supérieur à la moyenne sont éligibles pour être des CHs candidats durant la période en cours afin de garantir que seuls les nœuds avec une énergie suffisante sont sélectionnés comme CHs. Ensuite, la station de base exécute l'algorithme PSO pour trouver les K meilleurs CHs. Une particule est représentée comme une séquence d'ID des CHs candidats.

La phase de formation des clusters se déroule par la station de base après avoir généré les CHs à partir de la particule. Des clusters à deux niveaux seront formés dans le but d'améliorer le passage à l'échelle du réseau et de minimiser le nombre de CHs actifs au cours de chaque période. La station de base crée les clusters de premier niveau en affectant chaque nœud non-CH à un CH en fonction de la valeur RSSI pour le lien entre eux. Dans le cas de plusieurs CHs, le nœud deviendra membre du CH ayant la force RSSI maximale. Tout CH dans le premier niveau est appelé CH primaire "Primary Cluster Head (PCH)" et doit rester actif pendant toute la période sans qu'il passe en mode endormi. Le deuxième niveau est créé par la station de base en regroupant tous les nœuds non-CHs qui sont restés non regroupés à partir du premier niveau. La

station de base affecte chaque nœud non-CH du deuxième niveau à un nœud du premier niveau en fonction de la valeur RSSI. Un nœud du premier niveau qui a des membres du deuxième niveau est appelé CH secondaire "Secondary Cluster Head (SCH)" et il n'a pas besoin d'être actif pendant toute la période et il est mis en veille après avoir transmis à la fois ses propres paquets de données et les paquets de données de ses membres. Une fois la formation des clusters est terminée, un nœud du réseau peut être soit un PCH, soit un SCH, soit un membre du cluster "Cluster Member (CM)" ou un nœud sans statut (UN). Le processus de formation des clusters est illustré par la Figure 2.8.

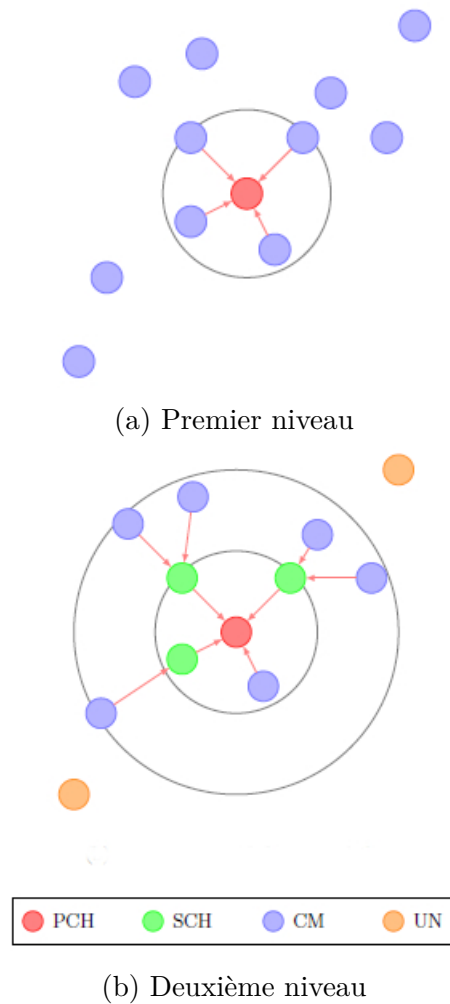


Figure 2.8: Processus de formation de clusters dans PSO-HC

La station de base crée un ordonnancement basé sur TDMA en allouant des créneaux horaires (time-slots) aux membres des clusters en fonction de la taille du cluster.

- *Phase steady-state*: Durant cette phase, chaque nœud membre utilise son échéancier TDMA pour transmettre ses données au nœud choisi comme prochain saut. Lorsqu'un nœud non-CH (un CM ou un SCH) termine son créneau de transmission de données, il entre en état de veille pour économiser son énergie.

PSO-HC a été évalué et comparé aux protocoles LEACH [24], LEACH-C [159] et PSO-C [158]. Les résultats obtenus ont montré que la consommation moyenne d'énergie par nœud dans PSO-HC est inférieure à celle que dans les autres protocoles et le débit est supérieur que dans les autres protocoles. Cependant, la complexité pour exécuter la phase set-up est importante en termes de temps et de messages échangés.

3. Le protocole MPSICA

"Multi-Particle-Swarm Immune Cooperative Algorithm (MPSICA)" [161] est un protocole de routage intelligent basé sur un algorithme PSO qui utilise un mécanisme de récupération rapide des routes. Ce protocole est conçu pour les RCSFs hétérogènes dont le but est de maintenir K chemins disjoints pour chaque nœud source à son CH et un chemin valide de chaque CH à la station de base. MPSICA est très tolérant aux pannes des chemins, puisque les retransmissions traditionnelles sont réduites et la fiabilité est améliorée avec une consommation d'énergie moindre et une durée de vie plus longue. Afin d'utiliser efficacement la consommation d'énergie, MPSICA est centralisé. Chaque nœud de MPSICA représente une particule où certains nœuds forment une séquence de particules qui correspond à un chemin vers la station de base. Ensuite, chaque séquence est évaluée en utilisant une fonction fitness pour trouver le chemin optimal. MPSICA est comparé aux protocoles EEHC [162] et ICE [163]. Les résultats ont montré que MPSICA consomme moins d'énergie par rapport à ces protocoles. Cependant, dans MPSICA l'overhead est important car il y a un grand nombre de messages qui sont échangés pour établir plusieurs chemins entre un nœud et son CH.

4. Le protocole TPSO-CR

Dans "Two-tier Particle Swarm Optimization protocol for Clustering and Routing (TPSO-CR)" [164], deux formulations de programmation linéaire (LP) aux problèmes de clustering et de routage sont représentées, suivies de deux algorithmes proposés pour les mêmes problèmes basés sur PSO. L'algorithme de clustering trouve l'ensemble optimal de CHs qui maximise l'efficacité énergétique, la qualité du cluster et la couverture du réseau. L'algorithme de routage est développé avec un nouveau schéma de codage de particules et une fonction Fitness pour trouver l'arbre de routage optimal qui relie ces CHs à la station de base. Ces deux algorithmes sont ensuite combinés en un protocole à deux niveaux pour fournir un modèle de clustering complet et pratique. L'effet de l'utilisation d'un réseau réaliste et d'un modèle de consommation d'énergie dans la communication basée sur les clusters pour RCSFs a été étudié. Des simulations approfondies sur 50 modèles de RCSFs homogènes et hétérogènes sont évaluées et comparées à des protocoles de réseau de capteurs basés sur des clusters bien connus. Les résultats ont montré que le protocole TPSO-CR fonctionne mieux que les protocoles LEACH [24], EHE-LEACH [165] et EEHC [162] en termes de diverses mesures de performance telles que le passage à l'échelle, le taux de livraison de paquets "Packet Delivery Rate (PDR)" au niveau des CHs et le nombre total de paquets délivrés à la station de base, et fournit des résultats légèrement similaires en termes de débit et consommation d'énergie à ceux des protocoles LEACH-C [159], PSO-C [158] et GA-C [166]. Cependant, une analyse théorique sur le nombre de CHs pourra illustrer mieux la qualité de l'approche qui détermine le

nombre de CHs dans ce travail.

5. Le protocole PSO-ECHS

Dans "PSO based Energy-efficient Cluster Head Selection algorithm (PSO-ECHS)" [167], les auteurs ont proposé un algorithme de sélection des CHs économe en énergie qui est basé sur l'optimisation de l'essaim de particules (PSO) appelé PSO-ECHS. L'algorithme est développé avec un schéma efficace de codage des particules et de fonction Fitness. Pour l'efficacité énergétique de l'approche PSO proposée, plusieurs paramètres ont été pris en considération tels que la distance intra-cluster, la distance de la station de base et l'énergie résiduelle des nœuds. La formation des clusters est faite de telle sorte que dans lesquelles les nœuds non-CHs rejoignent leurs CHs en fonction de leurs poids. L'algorithme a été testé de manière approfondie sur divers scénarios de RCSFs avec un nombre variable de nœuds et de CHs.

PSO-ECHS a été comparé aux protocoles LEACH [24], E-LEACH, LEACH-C [159], PSO-C [158] et LDC [168]. Les résultats ont montré que PSO-ECHS consomme moins d'énergie et prolonge mieux la durée de vie du réseau comparé à ces protocoles. Cependant, les auteurs n'ont pas prouvé que le nombre de CHs choisis est optimal.

6. Protocole proposé par Tukisi et al.

Tukisi et al. [169] ont proposé trois stratégies de routage écoénergétiques pour la conception d'un algorithme de routage basé sur l'optimisation de l'essaim de particules pour les RCSFs. La première stratégie maximise l'énergie du nœud ayant la plus faible quantité d'énergie dans le RCSF, la deuxième stratégie maximise l'énergie globale du RCSF et la troisième stratégie maximise l'énergie du nœud le moins performant tout en améliorant la latence du réseau. La meilleure de ces stratégies est ensuite testée dans un RCSF avec des nœuds de récupération d'énergie. Les résultats ont été comparés à une variante de protocoles: LEACH [24] et LEACH-CS (LEACH-Centralized Sleeping) [170]. Les résultats ont montré que la stratégie qui maximise l'énergie globale du réseau peut améliorer la durée de vie du réseau et ils ont montré également que le protocole de routage prenant en charge la récupération d'énergie peut prolonger la durée de vie d'un RCSF plus qu'un protocole qui ne prend pas en charge la récupération d'énergie. Cependant, les auteurs n'ont pas prouvé que le nombre de CHs est optimal à travers une étude théorique.

Optimisation par les colonies de fourmis

La métaheuristique d'optimisation par colonies de fourmis "Ant Colony Optimization (ACO)" a été initialement introduite par Dorigo et al. [171] pour résoudre les problèmes d'optimisation combinatoire difficiles. Cette métaheuristique est inspirée du comportement des fourmis qui communiquent entre elles en utilisant une substance chimique appelée phéromone. Lorsque les fourmis se déplacent, elles déposent du phéromone sur le sol pour marquer le chemin entre le nid et la source de nourriture. Une fourmi qui rencontre une trace de phéromone déjà déposée, elle peut la détecter et décider de la suivre avec une probabilité proportionnelle à l'intensité de la trace, et renforce ainsi cette trace avec sa propre phéromone, et en absence de phéromone la fourmi se déplace aléatoirement. La Figure 2.9 présente le fonctionnement simple de la colonie de fourmis (ACO) dans lequel les fourmis trouvent le meilleur chemin entre une source de nourriture et leur nid. Selon cette figure : (i) La première fourmi trouve la source de nourriture, en empruntant un

chemin quelconque (a), puis retourne à la fourmilière en laissant derrière elle des traces de phéromones (b), (ii) les fourmis empruntent indifféremment les quatre chemins possibles, mais la concentration de la phéromone rend le plus court chemin plus attractif, (iii) les fourmis empruntent le chemin le plus court, les portions longues des autres chemins perdent leur trace de phéromones en fonction du temps.

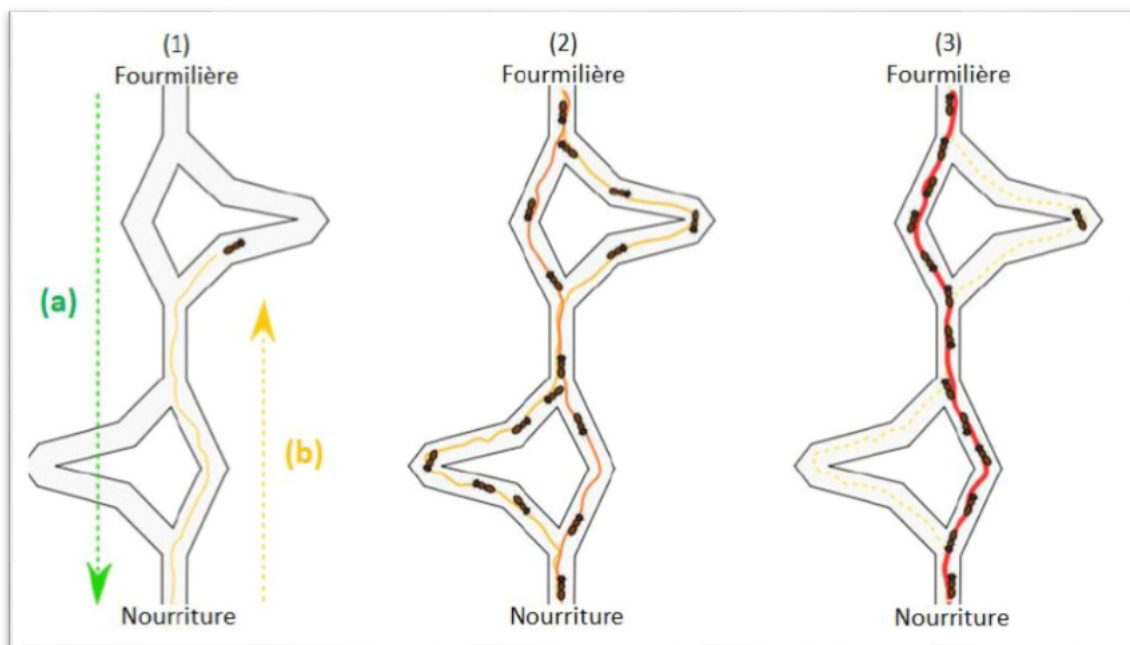


Figure 2.9: Comportement d'une colonie de fourmis [10]

L'idée principale de la métaheuristique ACO est de modéliser le problème comme un problème de recherche de chemins optimaux en construisant un graphe de chemins [133]. Plusieurs protocoles de routage basés sur la métaheuristique ACO ont été proposés dans la littérature pour améliorer les performances des RCSFs.

1. Le protocole T-ANT

T-ANT [172] est un protocole de routage utilisant le clustering dans lequel la formation des clusters est obtenue par la métaheuristique ACO tout en visant que le nombre de clusters formés soit optimal. T-ANT comprend deux phases. La première phase consiste en l'élection des CHs et la formation des clusters tandis que la seconde phase concerne l'estimation de la variance des données et la détection des redondances.

Dans T-ANT, pendant l'initialisation du nœud, la station de base libère un certain nombre de fourmis de telle sorte que le rapport du nombre de fourmis sur le nombre de nœuds devrait être égal à 0,1 (10%). Lorsque la station de base libère une fourmi, cette dernière choisit au hasard un de ses voisins. La fourmi pourrait pénétrer dans le réseau aussi profondément que restreint en fonction de son TTL (Time To Live). Lorsqu'une fourmi arrive à un nœud, le nœud suivant est choisi au hasard si son TTL n'a pas encore expiré. Si son TTL expire, la fourmi reste au niveau de ce nœud. Si toutefois l'emplacement final de la fourmi chevauche avec celui d'une autre fourmi, l'ancienne fourmi doit trouver un autre emplacement. Par ailleurs, la

phase de formation de clusters (CS) est contrôlée par un temporisateur. Lorsque ce temporisateur expire, un nœud vérifie s'il possède une fourmi. Si le nœud a une fourmi, il deviendra CH. Lorsqu'un nœud devient CH, il avise ses voisins en diffusant un message ADV contenant son identifiant (ID) et un champ TTL pour contraindre la propagation du message ADV. Lors de la réception d'un message ADV, un nœud normal enregistre l'ID du CH, l'ID de l'expéditeur comme parent, la distance en nombre de sauts vers ce CH, le nombre de messages ADV reçus jusqu'à présent et la distance totale exprimée en nombre de sauts vers tous les CHs à partir desquels il a reçu des messages ADV, puis rediffuse si son TTL le permet. Un nœud décide de rejoindre un cluster lorsque son temporisateur expire. Il calcule ensuite son niveau de phéromones en fonction de sa distance totale en nombre de sauts aux CHs, du nombre de CHs dans son voisinage et de son énergie résiduelle.

Un nœud non affilié choisit le meilleur cluster à rejoindre en fonction de sa distance en nombre de sauts au CH, ce qui garantirait une dissipation d'énergie minimale pendant les cycles de collecte de données. Le nœud rejoint un cluster en envoyant un message JOIN avec son identifiant, l'identifiant du CH sélectionné et son niveau de phéromone. Si le CH est à portée, le message est transmis directement; sinon, il est transmis par le biais de son parent au CH. Lorsqu'un CH reçoit des messages JOIN, il cherche le membre avec le niveau de phéromone le plus élevé pour attirer sa fourmi pour la phase CS suivante. Avant l'expiration du temporisateur CS, les fourmis se déplacent vers les nœuds ayant le niveau de phéromone le plus élevé parmi leurs voisins, et ces nœuds seront les CHs suivants. En outre, avant qu'une fourmi ne quitte son nœud actuel, une quantité d'anti-phéromone est déposée pour imiter une décroissance rapide du niveau de phéromone. Cela garantit que les fourmis ne reviennent pas au même nœud trop tôt, ce qui favorise l'équilibrage de charge.

Dans la phase steady-state, si un nœud est considéré comme redondant, il envoie ses données collectées à son CH correspondant. Il est possible que les fourmis meurent en raison de la défaillance d'un nœud. Ainsi, pour éviter une réduction du nombre de fourmis dans le réseau au fil du temps, la station de base relâche le même nombre optimal de fourmis pour redémarrer le processus. Par ailleurs, afin de déterminer les nœuds qui ont des informations redondantes et qui pourraient ainsi être mis en veille, un algorithme pour l'estimation de la variance des données et la détection des redondances a été introduit.

Les résultats de simulations ont montré que la moyenne de l'énergie consommée par round dans T-ANT est meilleure comparée à celle des protocoles TCCA [173] et "multihop-LEACH (m-LEACH)" [24]. Cependant, dans T-ANT l'overhead est important dans la phase d'élection des CHs et de formation de clusters.

2. Le protocole EBAB

"Energy Balanced Ant-Based routing protocol (EBAB)" [174] est un protocole de routage dynamique adaptatif basé sur l'optimisation par colonies de fourmis. EBAB vise à équilibrer la consommation d'énergie dans les RCSFs afin de prolonger leurs durée de vie. Dans EBAB, les clusters sont créés au début du processus de routage qui est divisé en deux schémas: intra-cluster et inter-cluster. Dans le schéma intra-cluster, l'algorithme est divisé en périodes et il comprend l'achèvement de l'élection des CHs où les CHs sont en concurrence dans les zones où ils appartiennent et

leur élection se fait en fonction de la puissance du signal avec la station de base. Après l'élection des CHs, les clusters sont formés et les CHs envoient un message dans leur rayon de transmission pour informer les autres nœuds. Le nœud transmet un message "ACK" pour rejoindre le cluster. Si le nœud reçoit plus d'un message, il choisira le CH en fonction de la distance et de l'énergie. Lorsque l'énergie d'un CH tombe en dessous d'une certaine valeur, ce dernier diffuse des informations sur la collecte de l'énergie résiduelle et de la distance des membres du cluster, puis calcule le coût (Fitness) de tous les nœuds, le nœud qui a le plus grand coût deviendra le successeur de ce CH. Dans la phase de transmission de données, au début de chaque période, tous les nœuds passent en mode actif. Ensuite, le CH diffusera les time-slots selon TDMA à ses membres. Chaque membre du cluster connaîtra son time-slot et gardera son émetteur-récepteur en mode éteint jusqu'à l'occurrence de son time-slot. Dans son time-slot, le membre du cluster transmet les données détectées à son CH correspondant. Dans EBAB, le processus du schéma inter-cluster utilise un algorithme ACO amélioré.

Le protocole EBAB a été comparé à LEACH [24] en termes de durée de vie du réseau et de quantité de données reçues par la station de base. Les résultats ont montré que la durée de vie dans EBAB est plus longue que celle de LEACH. Les résultats ont également révélé que la quantité de données reçues par la station de base dans EBAB est 1,7 fois supérieure à celle que dans LEACH.

3. Le protocole ACO-C

"Ant Colony Optimization for adaptive Clustering (ACO-C)" [175] est un protocole de routage adaptatif et économe en énergie pour les RCSFs basé sur la métaheuristique ACO. Ce protocole est centralisé au niveau de la station de base. Cette dernière sélectionne uniquement les nœuds dont l'énergie résiduelle est supérieure à l'énergie moyenne du réseau et utilise l'algorithme ACO pour trouver la meilleure solution selon la fonction coût. Après avoir trouvé la meilleure solution, la station de base informe chaque nœud de son CH et du cluster auquel il appartient. Chaque CH doit coordonner la transmission des données dans son cluster en attribuant un time-slot selon l'approche TDMA à chaque membre. A la fin de chaque période, chaque CH agrège les données envoyées par ses membres en un seul paquet et l'envoie à la station de base. L'algorithme ACO-C est évalué dans deux scénarios de réseaux où chacun contient 100 nœuds avec une énergie initiale inégale. Dans le premier scénario, la station de base est située au milieu du réseau, tandis que dans le second scénario, elle est située en dehors de la zone de déploiement des capteurs. Le protocole ACO-C est comparé à LEACH [24], LEACH-C [159] et PSO-C [158] dans les deux scénarios. Les résultats obtenus ont montré que la durée de vie du réseau dans ACO-C est plus longue comparé aux trois autres protocoles et que ACO-C surpasse LEACH, LEACH-C et PSO-C en termes de nombre de données reçues par la station de base.

4. Le protocole ACA-LEACH

"An Ant Colony clustering routing Algorithm (ACA-LEACH)" [174] est une version améliorée de LEACH qui exécute la métaheuristique ACO dans le mécanisme de routage inter-cluster pour réduire la consommation d'énergie des CHs et ainsi prolonger la durée de vie du réseau. L'élection des CHs dans ACA-LEACH dépend

de leur énergie résiduelle et de la distance qui les sépare des autres nœuds. Si la distance entre deux CHs est inférieure à un certain seuil, ACA-LEACH compare la valeur énergétique entre les deux, puis choisit le CH avec le plus d'énergie comme nouveau CH mais avec moins de nœuds membres. Dans ACA-LEACH, k fourmis sont placées au niveau de chaque CH en définissant une matrice qui est utilisée pour enregistrer les chemins générés. En outre, ACA-LEACH sélectionne le chemin optimal entre chaque CH et la station de base en se basant sur la valeur des phéromones. Ensuite, les k fourmis de chaque CH sont transférées vers le prochain CH accessible, avec la possibilité pour chaque CH adjacent qu'il soit sélectionné comme prochain saut. Les k fourmis dans chaque CH doivent choisir le chemin le plus court avec une consommation d'énergie minimale. Les résultats de simulations ont montré que le protocole ACA-LEACH consomme moins d'énergie et prolonge plus la durée de vie du réseau que LEACH [24].

5. Le protocole MRP

Le protocole "A Multi-path Routing Protocol based on clustering and ACO (MRP)" [176] est un protocole de routage multi-chemins basé sur un clustering dynamique utilisant ACO, conçu pour les RCSFs. MRP s'exécute en trois phases: la phase de formation des clusters, la phase de construction des multi-chemins et la phase de transmission de données. Lors de la phase de formation des clusters, les nœuds qui ont plus de chance pour devenir CH, ce sont les nœuds qui ont plus d'énergie, un nombre de voisins élevé et un indicateur RSSI élevé (supérieur à un certain seuil). Une version améliorée de l'algorithme ACO est utilisée afin d'établir plusieurs chemins avec des coûts énergétiques minimaux. Trois types de fourmis sont utilisés dans MRP: les fourmis de recherche "Search ANT (SANT)" dont le rôle est de trouver les chemins et de recueillir les informations le long de ces chemins jusqu'à la station de base; les fourmis arrières "Backward ANT (BANT)" utilisées pour évaluer le coût de chaque chemin découvert par les SANTs; et les fourmis anormales (AANT), utilisées pour éviter la défaillance du protocole. Si l'énergie résiduelle d'un CH est inférieure à 50% de la moyenne de l'énergie résiduelle de tous les nœuds membres du cluster, un nouveau CH sera élu. Le CH lancera un nouveau processus de découverte de chemins si le nombre de chemins est inférieur à deux, ce qui signifie que la fiabilité des chemins a considérablement diminué. L'évaluation des performances de MRP a été réalisée par l'utilisation de métriques comparatives telles que l'énergie moyenne, la consommation d'énergie, l'écart d'énergie et la durée de vie du réseau sur des réseaux dont la taille varie de 100 à 500 nœuds. Sur la base de ces paramètres, MRP est comparé à TEEN [8], à MP [177] et à MACS [178]. Les résultats de simulations ont montré que le protocole MRP donne une meilleure énergie moyenne et une faible consommation d'énergie comparé aux autres protocoles. L'écart-type de l'énergie dans le MRP est légèrement inférieur à celui de TEEN et les résultats ont également montré que MRP améliore la durée de vie du réseau nettement plus que les trois autres protocoles.

6. Le protocole IACO

"WSN routing method based on Improved Ant Colony Algorithm (IACO)" [179] est un protocole de routage basé sur une version améliorée de la métaheuristique ACO. Ce protocole permet d'établir et de mettre à jour la concentration des phéromones,

et choisir de manière adaptative une route de données en adoptant les avantages de l'état dynamique du réseau. Ensuite, les délais de transmission des données, le taux de perte de paquets et l'énergie résiduelle des nœuds sont déterminés en calculant une fonction de probabilité. Dans ce protocole, un chemin optimal est sélectionné en fonction du niveau de QoS. Le chemin optimal est le chemin qui contient le moins de sauts entre le nœud source et la station de base et implique les nœuds dont l'énergie résiduelle est la plus élevée. Pour maintenir le routage, des données sont transférées pour augmenter la concentration de phéromones. Les résultats de simulations ont montré que le protocole IACO réduit le délai de transmission des données, améliore l'efficacité du transfert de données, réduit également le taux de perte de paquets, réduit la consommation d'énergie et prolonge les performances globales du réseau et l'utilisation des ressources lorsqu'il est comparé aux protocoles suivants : protocole basé sur ACO, protocole utilisant Dijkstra et le protocole DEBA.

Les protocoles de routage basés sur la colonie des abeilles

L'algorithme de colonie d'abeilles artificielles "Artificial Bee Colony (ABC)" est une nouvelle métaheuristique introduite par Karaboga [180] et développée depuis 2005 par Karaboga et Basturk [181] pour la résolution des problèmes d'optimisation continue [182]. Cet algorithme est inspiré des comportements de recherche de nourriture des abeilles. Il considère une source de nectar comme une solution possible au problème à traiter. Par ailleurs, les abeilles ont des capacités individuelles proactives et peuvent s'auto-organiser [183]. La colonie d'abeilles artificielle est composée de trois types d'abeilles: les ouvrières, les spectatrices et les scoutes. Les abeilles scoutes quittent la ruche afin de détecter d'éventuelles sources de nourriture. Lorsqu'une source de nourriture est découverte, l'abeille scoute retourne dans la ruche pour recruter les abeilles ouvrières à travers une danse spéciale [184]. L'abeille ouvrière exploite la source de nourriture trouvée et l'abeille spectatrice attend le retour des ouvrières au champ de danse pour observer leurs danses et recueillir des informations sur les sources de nectar qu'elles ont trouvées. Les algorithmes basés sur ABC sont largement utilisés pour résoudre efficacement les problèmes NP-difficiles. Dans ce qui suit, on donne une description formelle d'un algorithme basé sur la métaheuristique ABC.

Dans ABC, chaque source de nourriture représente une solution candidate et la quantité de nectar de la source de nourriture représente l'adéquation de cette solution (fitness). Le nombre de sources de nourriture est égal au nombre d'abeilles ouvrières et d'abeilles spectatrices.

L'algorithme ABC standard génère initialement une population aléatoire de taille SN. Chaque source de nourriture représente une solution candidate pour le problème d'optimisation. La solution est généralement représentée par un vecteur de taille D où D est la dimension de la solution. La quantité de nectar de la source de nourriture simule l'adéquation de la solution (fitness). L'équation 2.8 est utilisée pour initialiser aléatoirement la population.

$$x_{ij} = l_k + rand(0, 1) * (u_k - l_k) \quad (2.8)$$

où x_{ij} est la nouvelle source, l_k est la limite inférieure de l'espace de solutions, u_k est la limite supérieure de l'espace de solutions, $rand$ est un nombre aléatoire dans la plage [0,1]. Le processus de recherche dans l'algorithme est exécuté par les abeilles "ouvrières".

Le nombre d'abeilles "ouvrières" est égal à la taille de la population SN. Le processus de recherche est répété jusqu'à un nombre prédéterminé de cycles appelé limite qui est l'un des paramètres efficaces d'optimisation de l'algorithme. Les abeilles "ouvrières" modifient continuellement la position de la source de nourriture en fonction de l'amélioration du fitness de la source de nourriture comme suit:

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \quad (2.9)$$

où ϕ_{ij} est un nombre aléatoire compris entre $[-1,1]$, v_{ij} est une nouvelle position d'une source de nourriture potentielle, x_{ij} est la position de la source de nourriture actuelle, x_{kj} est une position de source de nourriture voisine (choisie au hasard) avec $k = 1, 2, \dots, SN$ et $k \neq i$ pour tout $j \in 1, 2, \dots, D$. Si le coût de la solution v_{ij} est meilleure que x_{ij} , alors v_{ij} remplacera x_{ij} dans la population, sinon, x_{ij} sera conservé dans la population sans aucune mise à jour.

Une fois que les abeilles "ouvrières" ont terminé leur recherche et obtenu des informations sur les sources de nourriture, elles reviennent à la ruche et partagent des informations dans la zone de danse avec les abeilles spectatrices. Le nombre d'abeilles spectatrices est égal à la taille de la population SN. Les spectatrices évaluent l'aptitude de chaque source de nourriture et calculent la probabilité qu'elle choisisse la source de nourriture candidate voisine.

$$P_i = \frac{fit_i}{\sum_{n=1}^S Nfit_n} \quad (2.10)$$

où fit_i est la valeur de fitness de la solution i .

Si le fitness ne s'améliore pas pendant le nombre limite de cycles, ils abandonnent la source de nourriture et lancent aléatoirement la recherche d'une nouvelle source de nourriture par une abeille scoute en utilisant l'équation 2.8. ABC standard ne considère qu'une seule abeille "scoute" pour chaque génération pour rechercher une nouvelle source de nourriture au lieu de la source de nourriture abandonnée.

Dans cette section, on présente quelques protocoles de routage conçus pour les RCSFs et qui utilisent la métaheuristique ABC pour améliorer leurs performances.

1. Le protocole ABC-C

"Cluster-based WSN routing using Artificial Bee Colony algorithm (ABC-C)" [185] est un protocole de routage économe en énergie utilisant le clustering en se basant sur l'algorithme de colonie d'abeilles artificielles (ABC). Le protocole ABC-C est une variante de LEACH où la différence principale se trouve dans le processus de sélection des CHs. Dans ABC-C, le processus de sélection des CHs se base sur ABC et il prend en compte les distances des nœuds vers les CHs, les distances des CHs vers la station de base et l'énergie comme paramètres de sélection. Pour qu'un nœud soit un candidat pour le rôle de CH, il doit avoir un niveau d'énergie supérieur à un certain seuil. Après que la station de base reçoit les informations nécessaires sur ces paramètres, elle exécute le processus de sélection des CHs. Les CHs choisis sont responsables de la collecte des données captées par les nœuds membres de leurs propres clusters, de l'agrégation de ces données reçues en un seul paquet et de l'envoyer à la station de base. La communication intra-cluster est faite selon TDMA et la communication inter-cluster selon CDMA. Les résultats de simulations

ont montré que le protocole ABC-C améliore la quantité de données reçues par la station de base, la consommation d'énergie et la durée de vie du réseau comparé à LEACH [24] et PSO-C [158].

2. Le protocole Bee-Sensor-C

"An Energy-Efficient and Scalable Multipath routing protocol (Bee-Sensor-C)" [184] est un protocole de routage multi-chemins basé sur un clustering dynamique qui comprend principalement trois phases: la phase de formation des clusters, la phase de construction des communications multi-chemins et la phase de transmission des données. Ce protocole adopte un clustering dynamique orienté événement. La formation des clusters est déclenchée quand un événement se produit. Bee-Sensor-C ajoute un nouvel agent appelé "HiveHeader" dans la ruche pour chaque capteur. La responsabilité principale de "HiveHeader" est de prétendre que le nœud veut être un CH dans une zone d'événements lorsqu'il détecte un événement. Les nœuds à proximité de cet événement vont s'activer et ensuite ceux qui ont des informations sur cet événement deviennent membre de ce cluster. La construction des communications multi-chemins se déclenche lorsqu'un CH veut envoyer des données à la station de base. Le CH vérifie s'il y a des ouvrières appropriées avec n'importe quel nœud dans le cluster pour des communications multi-chemins existantes. La transmission de données se produit quand l'abeille scoute avec un identifiant de chemin unique arrive au CH et recrute des abeilles "ouvrières" en utilisant la danse du waggles. Bee-Sensor-C a été évalué dans deux scénarios différents. Dans le premier scénario, un seul événement s'est produit sur le réseau alors que dans le second scénario, plusieurs événements (au moins 2) peuvent être détectés. Il a été comparé aux protocoles "A Bee-Inspired power aware routing protocol (BeeSensor)" [186], "Improved Energy-Efficient Ant-Based Routing (IEEABR)" [187], et "Flooded Forward Ant routing (FF-Ant)" [188]. Les résultats de simulations ont révélé que Bee-Sensor-C consomme moins d'énergie avec un taux de livraison de paquets plus grand par rapport aux trois protocoles.

3. Le protocole BeeSwarm

BeeSwarm [189] est un protocole routage hiérarchique utilisant l'intelligence computationnelle (IC) basée sur les métaheuristiques. Ce protocole se compose de trois phases: BeeCluster (set-up), BeeSearch (découverte de routes) et BeeCarrier (transmission des données). Dans la phase BeeCluster, une approche similaire à l'approche standard ABC est utilisée pour la formation des clusters optimaux appelés "Bee Zones (BZs)" et pour la sélection des CHs correspondants appelés "Zonal Head (ZH)". Pour la formation d'une zone BZ, BeeSwarm applique un clustering hiérarchique agglomératif, dans lequel deux clusters qui sont séparés par une petite distance sont fusionner pour obtenir un cluster optimal afin de minimiser la consommation d'énergie. Les nœuds dans les zones d'abeilles (BZs) sont regroupés selon la fonction de liaison $D(X, Y)$, décrite comme suit:

$$D(X, Y) = \max_{x, y \in X, Y} d(x, y) \quad \forall X, Y \quad (2.11)$$

où x, y sont deux éléments qui appartiennent respectivement aux clusters X, Y . De même, la fonction de liaison complète $D(Z_i, Z_j)$ calcule la distance entre les nœuds

n_i, n_j appartenant respectivement aux zones Z_i, Z_j comme:

$$D(Z_i, Z_j) = \max_{n_i, n_j \in Z_i, Z_j} d(n_i, n_j) \quad \forall Z_i, Z_j \quad (2.12)$$

Après la création des BZs, des ZHs sont élus pour chaque BZ en fonction de leur énergie résiduelle et de la distance qui les sépare de la station de base. Après la fin de la phase BeeCluster vient la phase BeeSearch qui permet la découverte des routes par des abeilles scoutes. La phase BeeSearch utilise une recherche en avant pour explorer le réseau et une recherche arrière qui maintient des routes différentes entre la station de base et les nœuds. Une fois que les nœuds soient affiliés à des ZHs spécifiques, chaque ZH prépare un ordonnanceur TDMA et le transmet à ses nœuds enregistrés dans son cluster. BeeSwarm implique une heuristique basée sur la programmation linéaire (LP) pour optimiser la transmission de données durant la phase BeeCarrier. Les résultats de simulations ont montré que BeeSarm surpasse légèrement les protocoles MRP [176] et ERP [190] en termes de nombre total de paquets envoyés à la station de base, de consommation d'énergie et de durée de vie du réseau.

4. Le protocole ABC-SD

ABC-SD [191] est un protocole de routage basé sur un schéma de clustering impliquant la métaheuristique ABC pour minimiser la consommation d'énergie dans les RCSFs. ABC-SD est un protocole semi-distribué qui consiste en deux phases: la phase set-up et la phase "steady-state". Le clustering est réalisé par la station de base en tenant compte de l'énergie résiduelle et la position des nœuds et de leurs voisins, et le routage des données est réalisé par tous les nœuds de manière distribuée. Pendant la phase set-up, la station de base commence par établir l'ensemble des candidatures pour le rôle de CH en se basant sur la distance qui la sépare de ces nœuds et de leur énergie résiduelle. Après la construction de cet ensemble, l'algorithme ABC est exécuté afin d'affecter les nœuds à leurs CHs et de former les clusters. Lors de la deuxième phase, un mécanisme de sommeil/réveil utilisant TDMA affecte des time-slots aux nœuds membres au sein de chaque cluster. Ce mécanisme leur permet de transmettre les données recueillies à leur CH correspondant tout en évitant le problème d'interférences. La transmission des données au sein d'un cluster pourra être soit en un seul saut, soit en multi-sauts, et cela en fonction de la densité du cluster et de la localisation du CH. Après que le processus de clustering soit achevé, chaque membre du cluster lance le processus de découverte de routes afin de construire un chemin vers son CH. Pour illustrer les performances du protocole ABC-SD, les auteurs l'ont comparé avec LEACH [24], LEACH-C [159], PSO-C [158], PSO-HC [160], et ABC-C [185]. Les résultats de simulations ont révélé que le protocole ABC-SD a une moyenne d'énergie consommée par les nœuds plus faible et une efficacité énergétique plus élevée que les autres protocoles. Les résultats ont montré également que le protocole ABC-SD minimise le nombre de nœuds non regroupés, améliore le débit et maximise la livraison des données par rapport à ces protocoles.

b) Les protocoles de routage basés sur l'apprentissage automatique

L'apprentissage automatique "Machine Learning (ML)" est une science qui consiste à développer des algorithmes d'apprentissage, qui apprennent à résoudre une tâche à partir des expériences et des actes sans intervention humaine ni reprogrammation [192]. ML permet de rendre les processus informatiques plus efficaces, fiables et rentables. Le point fort du ML est sa capacité de fournir des solutions généralisées à partir d'une architecture capable d'apprendre à améliorer ses performances. En raison de la nature interdisciplinaire, ML joue un rôle primordial dans divers domaines, notamment l'ingénierie, le médical et l'informatique. ML a été appliqué pour résoudre divers problèmes dans les RCSFs [193].

Dans ce qui suit, nous présentons les protocoles de routage basés sur l'apprentissage automatique. Nous avons fait appels à quatre techniques utilisant le ML qui sont les plus utilisées dans la conception des protocoles de routage pour les RCSFs.

La logique floue La "Logique Floue (LF)" est une discipline mathématique inventée pour exprimer le raisonnement humain approximatif. Contrairement à la théorie classique des ensembles qui permet aux éléments d'appartenir ou non à un ensemble, LF permet de mesurer l'imprécision ou l'incertitude qui est marquée par l'utilisation de variables linguistiques comme "la plupart", "beaucoup", "fréquemment" par des règles au sein d'un ensemble appelé ensemble flou [133]. Un exemple d'une fonction d'appartenance floue pour la variable d'entrée énergie est présenté dans la Figure 2.10 où l'énergie est donnée par un ensemble de fonctions d'appartenance Low, Medium, High.

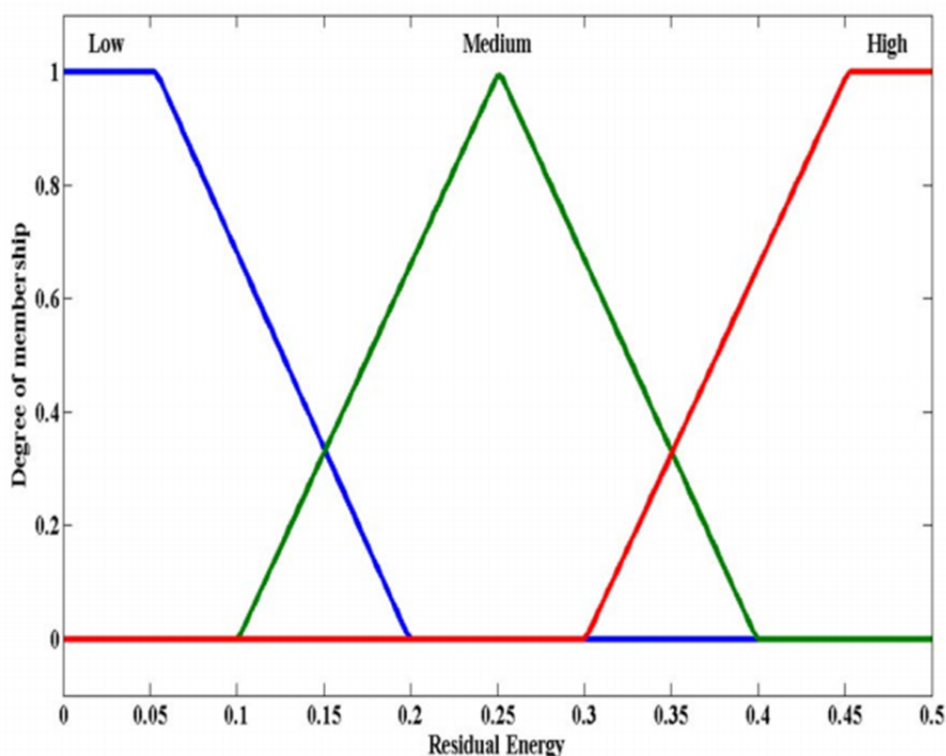


Figure 2.10: Exemple d'une fonction floue pour la variable d'entrée énergie [11]

Il existe plusieurs contributions basées sur la logique floue pour optimiser les protocoles

de routage et ainsi améliorer leurs performances.

1. Le protocole FCH

"Cluster-head election using Fuzzy Logic for WSNs (FCH)" [194] utilise la logique floue pour l'élection des CHs. Le choix des CHs dépend de l'énergie, la concentration du nœud et sa centralité durant le calcul. Le CH est élu par la station de base à chaque round en affectant une valeur pour chaque nœud qui définit sa chance de devenir CH, aidé par les descripteurs flous (énergie, concentration et centralité). Le modèle de la logique floue comprend quatre étapes : la fuzzification des descripteurs d'entrée, l'évaluation des règles, l'agrégation des règles et la défuzzification. Le protocole FCH a été évalué et comparé au fameux protocole LEACH [24]. Les résultats de simulations ont montré que FCH prolonge mieux la durée de vie du réseau par rapport à LEACH.

2. Le protocole CHEF

Le mécanisme d'élection des CHs dans "Cluster Head Election mechanism using Fuzzy logic (CHEF)" [195] est basé sur la logique floue. Contrairement à LEACH [24], CHEF utilise la distance vers la station de base et l'énergie restante des nœuds pour la sélection des CHs. A chaque round, un nœud génère une valeur aléatoire entre 0 et 1, puis la compare avec un seuil appelé "Popt". Si la valeur aléatoire est inférieure au seuil "Popt", la valeur de la chance est calculée en utilisant des règles floues IF-THEN. Ainsi, le nœud envoie un message appelé "Candidate_Message" contenant la valeur de la chance. L'envoi de ce message par un nœud signifie que ce nœud est candidat pour être CH. Le nœud qui a envoyé un message "Candidate_Message" attend tous les messages "Candidate_Message". Si sa propre valeur chance est supérieure aux autres valeurs, le nœud avertit le réseau avec un message "CH-message" pour s'annoncer comme CH. Si un nœud qui n'est pas un CH reçoit le message "CH-message", le nœud sélectionne le CH le plus proche pour être membre de son cluster en envoyant un message "Join-Cluster" au CH correspondant. La formation des clusters dans CHEF est plus efficace comparé à LEACH. Les résultats de simulations ont montré que la durée de vie du réseau dans CHEF est améliorée de 22,7 % par rapport à celle dans LEACH.

3. Le protocole LEACH-FL

Le protocole "improving on LEACH protocol of WSNs using Fuzzy Logic (LEACH-FL)" est une version améliorée de LEACH utilisant la logique floue [196]. Il se déroule de la même manière que LEACH seulement l'élection des CHs est centralisée. Les paramètres utilisés pour l'élection des CHs sont l'énergie résiduelle, la distance par rapport à la station de base et la densité des nœuds. Ces paramètres sont utilisés pour calculer la valeur de la probabilité d'un nœud pour devenir CH. LEACH-FL comprend trois parties principales: un module de fuzzification (quatre fonctions), un moteur d'inférences et un module de défuzzification. Les résultats de simulations ont montré que LEACH-FL est meilleur en termes de taux de consommation d'énergie et de durée de vie du réseau comparé à LEACH.

4. Le protocole ICT2TSK

"Improved Clustering algorithm using a Type-2 Takagi-Sugeno-Kang Fuzzy Logic System (ICT2TSK)" [197] est un algorithme de clustering amélioré qui utilise un

Takagi-Sugeno-Kang de type 2 (TSK) comme un système de logique floue. ICT2TSK permet l'élection des CHs au niveau de la station de base en choisissant ceux qui ont la possibilité de mieux gérer les règles d'incertitudes. Il équilibre la charge du réseau en introduisant un rayon de concurrence fixe pour chaque CH afin d'améliorer l'efficacité énergétique. Lorsque le réseau commence à fonctionner, tous les nœuds utilisent le protocole LEACH pour envoyer leurs informations (position et énergie résiduelle). Le système flou dans ce protocole permet de calculer la valeur qui définit pour chaque nœud la probabilité de devenir CH en fonction de son énergie résiduelle, de la distance qui le sépare de la station de base et du nombre de nœuds voisins. Les résultats de simulations ont montré que le protocole ICT2TSK surpasse LEACH [24], CHEATS [198] et EEUC [199] en termes de durée de vie du réseau et d'équilibrage de charge du réseau.

5. Le protocole SEP-FL

"Fuzzy logic approach to improve Stable Election Protocol (SEP-FL)" [200] est une version améliorée du protocole "Stable Election Protocol (SEP)" [153]. Ce protocole est basé sur la logique floue et vise à améliorer l'élection des CHs dans les RCSFs hétérogènes. Il permet d'offrir une période de stabilité plus longue et une période d'instabilité plus courte et augmente la durée de vie des nœuds. Ceci est réalisé grâce à l'implication du niveau de l'énergie de chaque type de nœuds et de leurs distances par rapport à la station de base dans le processus d'élection des CHs. Le système flou est divisé en deux systèmes d'inférences flous (SIF): un pour chaque type de nœuds (nœud avancé et normal). SEP-FL a été évalué et comparé avec LEACH [24], LEACH-LF [196], et l'algorithme original SEP [153], les résultats ont montré que SEP-FL prolonge la durée de vie du réseau et réduit la consommation d'énergie mieux que ces trois protocoles.

6. Le protocole EAUCF

Dans "Fuzzy Energy-Aware Unequal Clustering algorithm (EAUCF)" [201], un algorithme basé sur la logique floue utilisant un clustering non équilibré a été impliqué pour réduire la consommation d'énergie des CHs au sein des clusters, surtout les CHs qui sont proches de la station de base, ou ceux qui disposent d'une faible quantité d'énergie. Le système flou implémenté dans EAUCF utilise une règle de mapping "IF-THEN" pour traiter les incertitudes dans l'estimation de la portée des CHs. De plus, l'algorithme implique également un modèle probabiliste qui est utilisé pour l'élection des CHs provisoires. A chaque round, tous les nœuds doivent générer un nombre aléatoire entre 0 et 1 et le comparer avec un seuil prédéfini. Si le nombre aléatoire est supérieur au seuil, le nœud devient un CH provisoire, pour l'élection final des CHs. EAUCF prend en compte l'énergie résiduelle et la distance vers la station de base avant de calculer le rayon de concurrence. Les résultats de simulations ont montré que le protocole EAUCF surpasse LEACH [24], CHEF [195] et EEUC [199] en termes de l'épuisement de l'énergie du premier nœud "First Node Died (FND)", l'épuisement de l'énergie de la moitié des nœuds déployés et de la consommation d'énergie.

7. Le protocole DFCLC

"Distributed Fuzzy Logic-based Clustering algorithm (DFCLC)" [202] est un algorithme de clustering basé sur la logique floue qui est exécuté de manière distribuée

par les nœuds du réseau. DFCL considère le réseau comme un arbre dans lequel un nœud peut être une station de base, une racine (CH), un élément (nœud dans un cluster), un parent ou un fils. Un nœud parent est un nœud intermédiaire qui reçoit les données envoyées par un nœud fils à la racine. Afin de sélectionner les CHs de manière efficace, chaque nœud exécute la règle de la logique floue avec cinq paramètres d'entrée: l'énergie résiduelle, la distance qui le sépare des autres nœuds de l'arbre (centralité), la distance qui le sépare de la station de base, le nombre de sauts et le nombre de nœuds voisins (densité d'un nœud). Lors de l'exécution de la logique floue, seuls les nœuds qui ont une probabilité plus élevée d'être sélectionnés comme un nouveau nœud racine sont pris en compte. DFCL permet la tolérance aux pannes dans le réseau si un capteur épuise son énergie. Lorsqu'un nœud reçoit un message de découverte pendant la phase set-up, il stocke l'identifiant du nœud émetteur et désigne ce nœud comme étant le nœud le plus proche de la station de base; ainsi, si le réseau plante, un chemin alternatif peut être utilisé. DFCL a été implémenté dans le simulateur NS2 pour tester et comparer ses performances en termes de consommation d'énergie, de nombre de nœuds vivants, de durée de vie du réseau et de quantité de messages reçus par la station de base. Le protocole DFCL a été comparé à LEACH [24], ACAWT [203], FCH [194] et CHEF [195]. Les résultats des expérimentations ont montré que le protocole DFCL surpasse les autres protocoles en termes de toutes les métriques expérimentales mises en évidence.

8. Le protocole SIF

"Swarm Intelligence based Fuzzy routing protocol for clustered WSNs (SIF)" [204] est un protocole de routage qui utilise le clustering et qui est basé sur l'intelligence artificielle. Ce protocole prend en compte l'énergie résiduelle des nœuds, la distance des nœuds vers la station de base et la distance des nœuds vers les centres des clusters pour la sélection des CHs. SIF utilise un algorithme de clustering basé sur "FCM" pour regrouper tous les nœuds en clusters équilibrés où les CHs sont sélectionnés par un système d'inférences flou. SIF intègre une intelligence artificielle hybride appelée FA-SA qui est basée sur l'algorithme de comportement des lucioles (FA) et sur l'algorithme de recherche locale recuit simulé (SA) qui est utilisé pour optimiser la table de la base des règles floues du système. Les auteurs ont démontré que le protocole SIF est efficace en énergie, en termes de formation de clusters équilibrés, de réduction de distances intra-cluster, de prolongation de durée de vie du réseau et de maximisation du nombre total de paquets de données reçus par la station de base. Les résultats de simulations ont montré que SIF prolonge mieux la durée de vie du réseau comparé aux protocoles LEACH [24], LEACH-DT [147], ASLPR [146] et LEACH-FL [196].

9. Le protocole FBUC

"Fuzzy Based Unequal Clustering for WSNs (FBUC)" [205] est un algorithme de clustering distribué qui se concentre sur la méthode de jonction des membres du cluster avec leur CH correspondant. FBUC est une version améliorée du protocole EAUCF [201]. Il utilise une valeur seuil probabiliste au lieu d'une valeur seuil prédéfinie à laquelle il ajoute une variable floue appelée degré de nœud pour l'élection des CHs pendant la compétition de portée comparé à EAUCF. Les membres rejoignent le CH en fonction de la distance et du degré de CH afin d'utiliser

efficacement l'énergie et d'augmenter la durée de vie du réseau. FBUC a été évalué et comparé aux protocoles LEACH [24] et EAUC [201]. FBUC montre une meilleure consommation d'énergie et une meilleure durée de vie du réseau que les deux protocoles.

10. Le protocole EEDCF

"Energy-Efficient Distributed Clustering algorithm based on Fuzzy approach (EEDCF)" [206] est un algorithme de clustering distribué basé sur la logique floue. L'approche proposée définit quatre états différents pour chaque nœud : l'état initial, l'état CH candidat, l'état CH élu et l'état du nœud membre. Pendant la première phase, chaque nœud doit construire sa propre table d'informations qui contient son ID, son énergie résiduelle, les ID de ses voisins, et leur énergies restantes correspondantes. Après chaque round, le nœud met à jour ses informations à l'aide du paquet "Node_MSG". Lors de la deuxième phase, le nœud effectue l'analyse en logique floue en fonction de son énergie résiduelle, du nombre de ses voisins dans son rayon de communication (degré du nœud) et de l'énergie résiduelle moyenne de ses voisins. A la fin de cette phase, chaque nœud se transforme en état "CH candidat" et envoie cette information (sa sortie) à tous ses voisins dans son rayon de communication. Lorsque la sortie est obtenue par des règles "IF-THEN" conformément au mécanisme du modèle flou "Takagi, Sugeno and Kang (TSK)" au lieu du modèle traditionnel, le nœud ayant la sortie la plus faible bascule vers l'état "nœud membre" et attend de rejoindre un cluster approprié après l'élection des CHs. Cependant, le nœud ayant la sortie la plus élevée devient CH et bascule vers l'état "CH élu". Les nœuds rejoignent les CHs appropriés en fonction du RSSI et construisent ensuite des clusters. Afin d'évaluer le protocole EEDCF, les auteurs ont examiné deux scénarios avec la même superficie et le même emplacement de l'application statistique. Le premier scénario déploie 100 nœuds tandis que le second utilise 150 nœuds dans le champ d'étude. Les expérimentations ont montré que EEDCF surpasse EADEEG [207] et DFLC [202] en termes de durée de vie et de débit de transmission des données.

Les protocoles de routage basé sur l'apprentissage par renforcement

L'"Apprentissage par Renforcement (AR)" en anglais "Reinforcement Learning" est un sous-domaine de l'apprentissage automatique qui enseigne à un agent ce qu'il doit faire et comment assigner des situations à des actions particulières afin d'être intelligent [208]. L'agent essaie plusieurs actions, et apprend de son expérience quelle est la meilleure action qu'il doit choisir afin d'optimiser les performances du réseau [133, 208–210]. L'apprentissage par renforcement est utilisé par les protocoles de routage dans les RCSFs pour trouver les meilleurs chemins et prolonger leur durée de vie de ces réseaux [211].

1. Le protocole CLIQUE

Le protocole "Role-free clustering with Q-learning for WSNs (CLIQUE)" [212] est un protocole de routage basé sur un clustering libre qui utilise un algorithme de l'AR appelé Q-Learning. Dans ce protocole, les nœuds doivent connaître l'identité du cluster auquel ils appartiennent. Contrairement aux approches traditionnelles de clustering utilisées, CLIQUE ne se sert pas des coûts des nœuds pour sélectionner les

CHs, mais utilise plutôt une stratégie du Q-Learning. L'AR appliqué dans CLIQUE permet aux nœuds d'apprendre et de décider indépendamment s'ils doivent ou non agir comme CH. Le CH doit avoir le meilleur coût de routage vers toutes les stations de base comparé aux autres membres de son cluster. CLIQUE attribue des Q-valeurs à chaque action possible où le processus d'apprentissage permet à l'agent (nœud) de choisir et d'exécuter une action (routage) et de recevoir ensuite la valeur correspondante utilisée pour actualiser la Q-valeur. CLIQUE a été comparé au protocole TRC [213], ce dernier a fourni une probabilité a priori à chaque nœud pour décider s'il peut devenir CH ou non. Les résultats obtenus après simulations ont montré que CLIQUE a de meilleures performances que TRC en termes de consommation d'énergie "FND", de livraison de données et d'overhead.

2. Le protocole QSGrd

"Q-Smart Gradient based routing protocol (QSGrd)" [214] est un protocole de routage hybride, qui combine Q-learning et le gradient de transmission, pour optimiser la consommation d'énergie dans les RCSFs. La probabilité de réussite de la transmission dépend uniquement de la distance entre les nœuds et de la portée de transmission maximale. Dans QSGrd, la composante Q-Learning prend en compte à la fois le nombre moyen estimé de transmissions vers la station de base par chaque nœud (déterminé par le gradient de transmission établi), ainsi que le niveau d'énergie résiduelle de chaque nœud. De cette façon, le protocole sera capable d'apprendre sur les routes et leurs longueurs entre chaque nœud et la station de base. Par conséquent, il pourra choisir la route dont la longueur est minimale entre chaque nœud et dont l'énergie résiduelle des nœuds le long de cette route est maximale.

Dans QSGrd, chaque nœud conserve une Q-valeur pour lui-même (qui est incluse dans les paquets de trame d'état qui sont diffusés aux voisins). Cette Q-Valeur est une mesure de l'utilité de ce nœud, à en juger par les métriques de la fonction de récompense Q-Learning. Au début, toutes les Q-valeurs sont initialisées à 1. Une fois que l'apprentissage commence, les Q-valeurs sont déterminées en fonction des opérations mathématiques spécifiées par les équations 2.13, 2.14, 2.15, 2.16 et 2.17. L'équation 2.17 détermine la valeur de récompense pour les progrès réalisés vers la station de base. De même, l'équation 2.16 détermine la valeur de récompense pour l'amélioration de l'énergie. Dans 2.16, les actions (a_n) sont récompensées pour avoir réduit le nombre de transmissions restantes pour que le paquet atteigne la station de base. Dans 2.16, les actions (a_n) sont récompensées pour l'augmentation de l'énergie résiduelle du nœud contenant le paquet. Pour 2.16 et 2.17, une fonction sigmoïde est appliquée de sorte que: i) les récompenses sont toujours positives, et ii) il y a un comportement de gains et de pertes décroissants lorsque les valeurs delta s'éloignent de 0. L'équation 2.15 effectue une combinaison pondérée des équations 2.16 et 2.17 permettant l'ajustement de quelle variable (énergie ou progression) est privilégiée. L'équation 2.14 est la fonction de récompense actualisée, au cœur de Q-Learning. Le terme $P_{suc}(a_n)$ est incorporé dans l'équation 2.14 afin que les valeurs de récompense soient échelonnées en fonction de la réalisabilité réelle de l'action correspondante. L'équation 2.13 applique le taux d'apprentissage.

$$Q_s = (1 - \alpha) \times Q_s + \alpha \times Q_{new} \quad (2.13)$$

$$Q_{new} = (\gamma \times Q_n + (1 - \gamma) \times R(a_n)) \times P_{suc}(a_n) \quad (2.14)$$

$$R(a_n) = \omega_e \times R_e(a_n) + (1 - \omega_e) \times R_p(a_n) \quad (2.15)$$

$$R_e(a_n) = \text{sigm}(\Delta E(a_n) \times \pi \times \beta_e) \quad (2.16)$$

$$R_p(a_n) = \text{sigm}(\Delta T_{est}(a_n) \times \pi \times \beta_p) \quad (2.17)$$

où α est le taux d'apprentissage, γ est le facteur de récompense actualisé, Q_s est la Q-valeur du nœud de transfert "s", Q_n est la Q-valeur du voisin "n" considéré, a_n est l'action de transmettre le paquet au nœud "n", $P_{suc}(a_n)$ est la probabilité d'accomplir avec succès l'action a_n , ω_e est le poids des considérations énergétiques, $\Delta E(a_n)$ est la différence de niveau d'énergie entre le nœud "s" et le nœud "n" où le niveau d'énergie est une valeur comprise entre 0 et 1, $\Delta T_{est}(a_n)$ est la différence entre le nœud "s" et le nœud "n" dans le nombre estimé de transmissions vers la station de base tel qu'observé dans le gradient du réseau, β_e est un facteur d'échelle de différence d'énergie, et β_p est un facteur d'échelle de différence de transmission.

Par ailleurs, QSGrd associe des probabilités de réussite de la transmission aux voisins d'un nœud par le biais du gradient de transmission. Ensuite, ces probabilités sont utilisées pour mettre à jour les Q-valeurs. Pour développer QSGrd, trois autres protocoles de base ont été implémentés "Greedy Gradient based routing protocol (GGrd)", "Energy Aware Gradient based routing protocol (EAGrd)" et "Energy Smart Gradient based routing protocol (ESGrd)". Les résultats de simulations ont montré que le protocole QSGrd a amélioré la durée de vie du réseau et a minimisé le délai de transmission comparé à ces trois protocoles.

3. Le protocole RLBR

"Reinforcement-Learning-Based Routing protocol (RLBR)" [215] est un protocole de routage qui est basé sur l'apprentissage par renforcement (AR). Il utilise l'AR pour rechercher le chemin de routage optimal pour la transmission des données. Ce dernier prend en compte la distance, l'énergie résiduelle et le nombre de sauts comme paramètre pour choisir le prochain saut. Dans RLBR, chaque nœud est considéré comme un agent. Pour tout nœud "i", lorsqu'il génère ou reçoit un paquet, l'état du paquet est le nœud "i", et l'action du nœud "i" est de sélectionner un nœud voisin "j" comme nœud relais en fonction du $Q(i, j)$ actuel. $Q(i, j)$ dénote l'évaluation de la qualité du chemin depuis nœud "i" à la station de base via le nœud "j". Après avoir déterminé le relais suivant, le nœud "i" met à jour sa propre Q-valeur en fonction du $Q(i, j)$ correspondant et met ses dernières informations dans l'en-tête du paquet. Une fois le paquet envoyé au nœud suivant, le nœud précédent du nœud "i" peut également écouter ce paquet, et la Q-valeur dans ce paquet peut être considérée comme une rétroaction du nœud "i" au nœud précédent. Pour le nœud précédent, la rétroaction peut être utilisée dans le prochain cycle de transmission de données. Grâce à l'apprentissage constant, la Q-valeur estimée de chaque chemin se rapproche de la valeur réelle.

Dans RLBR, le réseau est initialisé par la station de base et au cours de cette phase, chaque nœud envoie un paquet de contrôle à ses nœuds voisins qui contient son emplacement, son énergie résiduelle et le nombre de sauts vers la station de base. Une fois un paquet de contrôle est reçu par le nœud "i", ce dernier extrait les informations de l'expéditeur et calcule la Q-valeur de l'expéditeur selon l'équation 2.18. Ensuite, les informations de l'expéditeur sont enregistrées dans la table des voisins du nœud "i". Avant d'envoyer le paquet de contrôle, le nœud "i" calcule son nombre de sauts à la station de base par l'équation 2.19 et met ses propres informations dans le paquet de contrôle.

$$Q(s) = \frac{E(s)}{h(s)} \quad (2.18)$$

$$h(i) = h(s) + 1 \quad (2.19)$$

où $E(s)$ est l'énergie actuelle de l'expéditeur et $h(s)$ est le nombre de sauts de l'expéditeur à la station de base.

Après cette phase, les paquets de données sont transmis dans le réseau. Les résultats de simulations ont montré que le protocole RLBR optimise la durée de vie du réseau et améliore l'efficacité énergétique comparé aux protocoles BEER [216], Q-Routing [217] et MRL-SCSO [218].

Les protocoles de routage basés sur les réseaux de neurones

Un "Réseaux de Neurones Artificiels (RNA) est une technique qui utilise un apprentissage supervisé basé sur le modèle d'un neurone humain pour classifier les données [219, 220]. Cette technique contient un grand nombre de neurones connectés (unités de traitement) qui traitent l'information et produisent des résultats précis. RNA fonctionne typiquement sur des couches. Ces couches sont connectées avec des nœuds et chaque nœud est associé à une fonction active. La Figure 2.11 montre la structure de base d'un RNA.

Chaque RAN contient trois types de couches appelées couches d'entrée, une ou plusieurs couche(s) cachée(s) et couches de sortie. RAN classifie très facilement des ensembles de données complexes et non linéaires, et il n'y a aucune restriction pour les entrées comme pour les autres méthodes de classification. Plusieurs applications des RCSFs temps réel ont utilisé RAN bien qu'il ait un besoin de calcul très élevé. RAN peut être appliqué pour améliorer l'efficacité de divers problèmes dans les RCSFs.

1. Le protocole de Cordina et al.

L'algorithme proposé par Cordina et al. [221] présente une nouvelle approche de clustering basée sur un réseau neurones de type "Self Organizing Map (SOM)". Cet algorithme vise à optimiser le réseau en le divisant en clusters efficaces afin de minimiser la consommation d'énergie. Il exécute des mécanismes qui incluent une valeur minimale de distance de séparation par rapport au CH. La stratégie SOM est utilisée pour l'élection des CHs à travers un système de rotation des CHs et d'équilibrage de charge aidé par des fonctions de coût. Chaque round commence par une phase set-up au cours de laquelle les clusters sont organisés. Lors de la phase

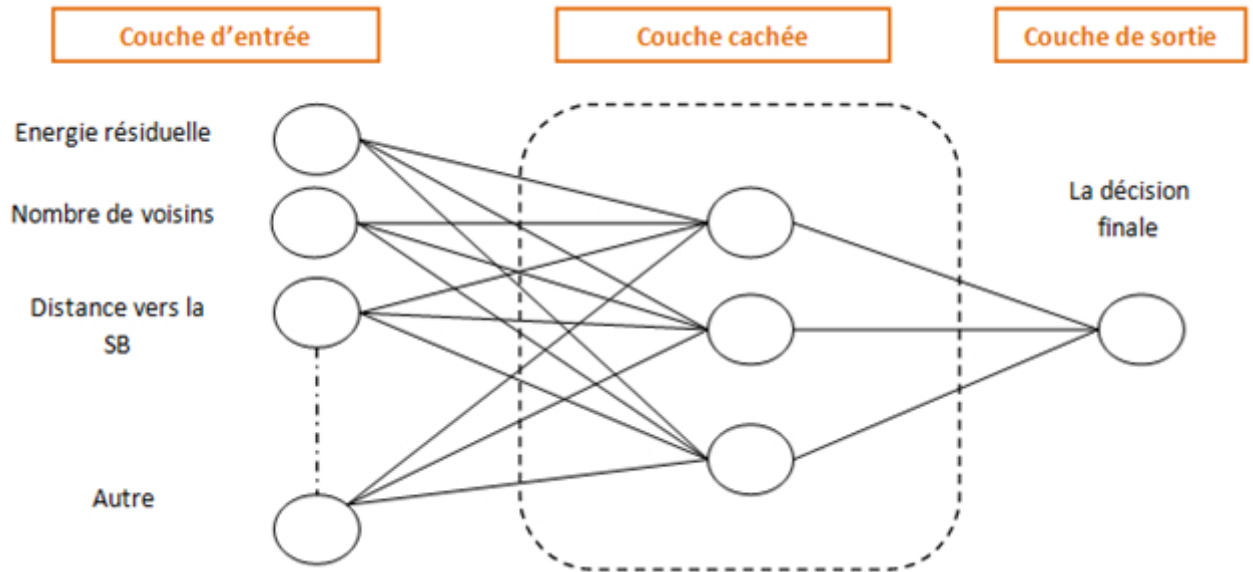


Figure 2.11: Un modèle de base d'un RNA

de transmission, les nœuds peuvent transmettre les données ou non au CH selon les scénarios. Les données reçues par le CH sont agrégées et transmises à la station de base. L'élection des CHs est effectuée par SOM à l'aide d'une cartographie des neurones à 4 entrées et 16 sorties, mais les nœuds et les CHs qui ont un niveau énergétique inférieure à un seuil critique envoient un message "Node dead"/"CH dead" directement à la station de base. Les résultats de simulations ont montré que le protocole proposé est meilleur que LEACH [24] en termes de durée de vie du réseau et de quantité de données délivrées à la station de base.

2. Le protocole de Nehra et al.

Nehra et al. [222] ont proposé un protocole de routage basé sur le clustering utilisant les réseaux de neurones pour optimiser la durée de vie des réseaux de capteurs. Le protocole proposé utilise une formulation de programmation linéaire (PL) pour choisir quelques nœuds avec un niveau énergétique élevé pour former une route optimale, tout en minimisant le coût total de la route. L'élection des CHs se fait en utilisant l'apprentissage adaptatif dans les réseaux de neurones, suivi par le routage et la transmission de données. Contrairement à [221], la stratégie proposée est définie en trois phases : la phase set-up, la phase de routage et la phase de transmission des données. Dans la phase set-up, les CHs peuvent être sélectionnés en fonction de la métrique de coût définie dans l'équation 2.20. Les parties peuplées du réseau (contiennent un grand nombre de nœuds) seront surpeuplées de CHs, tandis que les zones à peine couvertes seront laissées sans aucun CH. Dans une telle situation, il est probable que les capteurs à coût élevé provenant de zones mal couvertes devront effectuer des transmissions de données coûteuses vers des nœuds CHs distants, réduisant ainsi leur durée de vie.

$$R_C = \frac{E_i^D}{E_t(S_i, S_j) + E_r(S_i, S_j)} \quad (2.20)$$

Où E_i^D est l'énergie associée au rapport de livraison de paquets provenant du nœud source S_i et correctement reçus par le nœud de destination, tandis que $E_t(S_i, S_j)$ est l'énergie utilisée pour transmettre de S_i à S_j et $E_r(S_i, S_j)$ est l'énergie dissipée pour recevoir le paquet.

Pendant la phase de routage et de transmission des données, le protocole proposé analyse tous les chemins à partir d'un CH et trouve le meilleur chemin pour atteindre la station de base. Le routage des données de chaque CH vers la station de base est effectué sur des chemins en mode multi-sauts, ce qui est donné en minimisant l'équation 2.20. Ainsi, un nœud source choisira le chemin le plus efficace en énergie afin de maximiser la durée de vie du réseau. Les résultats de simulations ont montré que le protocole proposé fournit de meilleures performances en termes de pourcentage de nœuds vivants comparé à PEACH [223].

3. Le protocole ELDC

Le protocole ELDC [224] est un protocole de routage économe en énergie et robuste basé sur RNA pour la surveillance de l'environnement. Ce protocole atteint ses objectifs au moyen d'un RNA qui est essentiellement baptisé sur des algorithmes arithmétiques basiques capables d'apprendre les mappages compliqués entre les entrées et les sorties. ELDC est une extension du protocole EEUC [199] et du protocole EEMDC [225] dans laquelle les auteurs ont tiré profit des avantages des deux protocoles et éviter leurs limites pour réduire la consommation d'énergie et ainsi prolonger la durée de vie du réseau. Dans ce protocole, un nouveau type de nœud dans le réseau est défini appelée "Chief Node". Lors du lancement du réseau les nœuds ayant le plus de voisins deviennent "Chief Node". De plus, il utilise le RNA pour sélectionner les CHs et de réduire le temps de sélection requis réalisant ainsi la tâche souhaitée en prenant l'énergie résiduelle du nœud, son nombre de voisins, sa distance vers le CH, sa distance vers "Chief Node", la charge du trafic, le total des interférences "Signal to Noise Ratio (SNR)" et l'état du lien de communication comme paramètres dans la couche d'entrée. Les résultats de simulations ont montré que ELDC surpasse LEACH [24], HEED [226], DDAR [227], EEUC [199], et EEMDC [225] en termes de consommation d'énergie et durée de vie du réseau.

Les protocoles de routage basés sur K-Means

K-Means [23] est une méthode d'apprentissage non supervisée pour résoudre uniquement le problème de clustering. K-Means permet de classer un ensemble de données en un nombre prédéfini de clusters fixé à l'avance. L'exécution de la méthode K-means commence par déterminer K centroïdes, un pour chaque cluster. Ensuite chaque nœud choisit le centroïde le plus proche et s'associe à son cluster. Une fois que tous les nœuds du réseau soient attachés à un cluster, cette étape se termine et des clusters sont créés. Après, K nouveaux centroïdes sont déterminés de nouveau en calculant les barycentres des clusters résultant de l'étape précédente. Après avoir obtenu ces K nouveaux centroïdes, une nouvelle liaison doit être établie entre les nœuds du réseau et les nouveaux centroïdes comme montré dans la Figure 2.12.

La complexité temporelle de l'algorithme K-Means est $\theta(N * K * i * d)$, où n représente le nombre de nœuds, K indique le nombre de centroïdes, i indique un nombre d'itérations, et d représente le nombre d'attributs. La fonction de minimisation de la somme des carrés

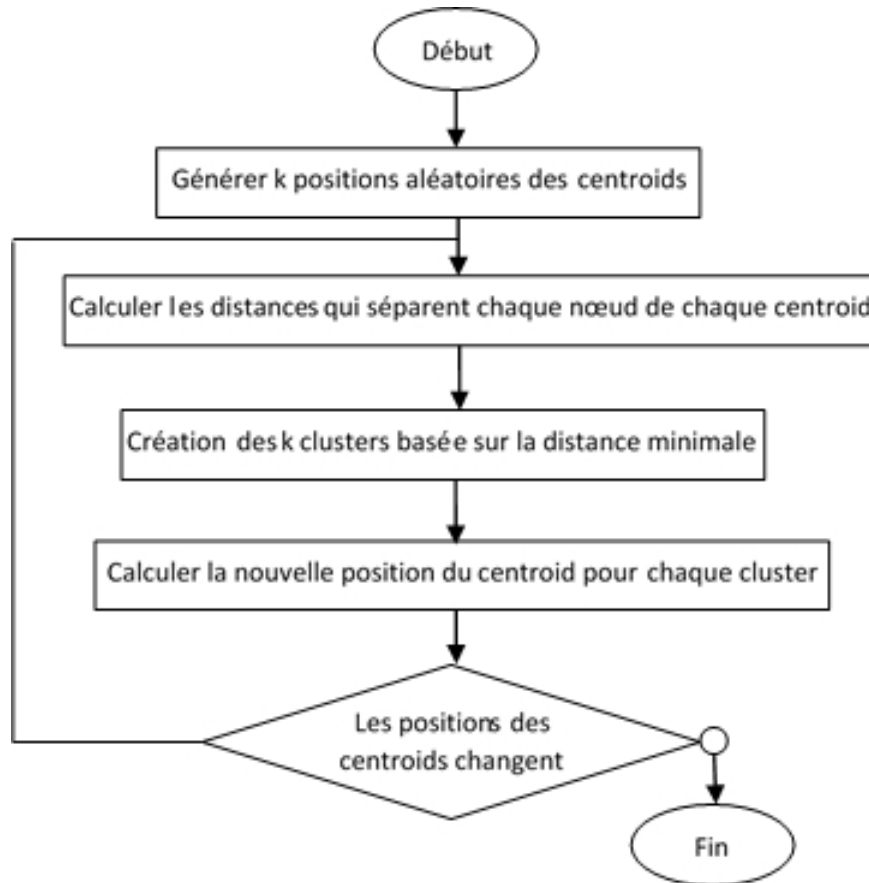


Figure 2.12: Étapes d'exécution de K-Means

des erreurs [228] est présentée par l'équation 2.21.

$$\text{Min } f(x) = \sum_{r=1}^K \sum_{(x_i \in C_r)} \|x_i - g_r\|^2 \quad (2.21)$$

où $\|x_i - g_r\|^2$ est la distance Euclidienne entre l'individu x_i et le centre du cluster g_r , C_r est la classe numéro r , x_i est un individu dans un cluster, et g_r est le centre de classe C_r .

K-Means est l'approche de clustering la plus simple et la plus adaptée aux RCSFs pour les structurer en clusters et trouver les bons cluster-heads pour acheminer les données vers la station de base.

1. Le protocole de Sasikumar et Khara

Dans [229], les auteurs ont proposé deux approches de clustering basées sur K-Means: une centralisée et l'autre distribuée.

- L'approche centralisée: Lorsqu'une autorité centralisée prend des décisions et partitionne les nœuds en clusters sans l'implication d'autres nœuds. Cette autorité centralisée obtient les informations nécessaires pour le clustering à partir des nœuds individuels. Sur la base de ces informations, il se regroupera par un algorithme et enverra les résultats du clustering aux nœuds individuels. Dans

cette approche, les CHs sont élus en fonction de leurs distances du centroïde et de leurs énergies résiduelles. Ensuite, une fois que le nœud central termine le processus de clustering et de sélection des CHs, ce dernier renvoie les informations à chaque nœud individuellement pour lui informer de son cluster et de son CH correspondant.

- L'approche distribuée: Dans cette approche, chaque nœud participe à la prise de décisions de clustering en obtenant les informations nécessaires pour le clustering à partir de tous les autres nœuds. Sur la base de ces informations, tous les nœuds seront regroupés par un algorithme et décideront également lesquels peuvent devenir des CHs. Étant donné que l'algorithme K-Means est basé sur les distances euclidiennes et les énergies (pour choisir les CHs), les informations sur les positions et les énergies de tous les nœuds sont obtenues par chaque nœud en échangeant des messages entre eux. Après avoir obtenu les informations sur tous les nœuds, chaque nœud exécute l'algorithme K-Means. L'algorithme K-Means pour le clustering et l'algorithme pour le choix des CHs sont similaires aux algorithmes utilisés dans le clustering centralisé. Comme chaque nœud exécute le même algorithme, chaque nœud sait à quel cluster il appartient et quel est son CH.

Ces deux approches ont été développées sous NS-2 et les résultats de simulations ont montré que dans l'approche distribuée le processus de clustering converge rapidement comparée à l'approche centralisée alors que la consommation moyenne d'énergie par nœud est presque la même dans les deux approches.

2. Le protocole LEACH-KED

Dans [230], un protocole de routage de clustering amélioré appelé "Clustering routing algorithm based on energy and distance (LEACH-KED)" a été proposé. LEACH-KED comprend deux phases: la phase de formation des clusters et la phase de stabilité. Dans ce protocole, la première phase est exécutée après chaque round en appliquant la méthode K-Means et en fixant K (le nombre de clusters) à 5% du nombre total des nœuds dans le réseau pour élire les CHs. L'élection des CHs est basée sur un poids qui dépend de l'énergie, de la distance par rapport au centroid et de la distance par rapport à la station de base. La deuxième phase concerne la formation des clusters et la transmission des données. La procédure de ces deux sous-phases est similaire à celle de LEACH [24]. Les résultats de simulations ont montré que LEACH-KED surpasse LEACH [24] et LEACH-C [159] en termes de consommation d'énergie.

3. Le protocole KEAC

Dans [231], les auteurs ont proposé un protocole de routage centralisé pour les RCSFs utilisant la méthode de clustering K-Means, appelé "K-means based Energy Aware Clustering algorithm (KEAC)". KEAC vise à réduire la consommation d'énergie sur l'ensemble du réseau. Dans KEAC, les CHs sont élus en fonction de leur énergie et de la distance qui les sépare de leurs centroïdes. Les CHs générés sont répartis uniformément dans le réseau en fonction de leur poids. Les communications intra-cluster et inter-cluster utilisent un mode de transmission à un seul saut en ajustant la portée de transmission. Néanmoins, l'ajustement de la portée de transmission pour

communiquer avec la station de base peut augmenter la surcharge de ce protocole. Le protocole KEAC a été évalué et comparé aux protocoles LEACH [24] et LEACH-C [159]. Les résultats de simulations ont montré que KEAC consomme moins d'énergie et prolonge plus la durée de vie du réseau par rapport à LEACH et LEACH-C.

4. Le protocole de Rabiaa et al.

Rabiaa et al. [232] ont proposé une nouvelle version améliorée du protocole LEACH. Ce protocole de routage proposé consiste à intégrer un algorithme amélioré non supervisé qui est K-means pour former les clusters et à utiliser l'algorithme d'élimination gaussien pour choisir les CHs. Dans ce protocole, sur la base de l'indice "Davies-Bouldin (DBI)", K-means trouve le nombre optimal de clusters (K). L'indice Davies-Bouldin rend les calculs beaucoup moins complexes, ce qui est un objectif potentiel pour tout algorithme de clustering. Ainsi, K-means basé sur DBI permet un calcul distribué et parallèle plus simple pour déterminer K et permet une classification automatique plus rapide des nœuds. De cette manière, des clusters équilibrés en termes d'énergie sont créés et la consommation d'énergie distribuée est garantie. Les résultats de simulations ont montré que le protocole proposé réduit considérablement le temps de calcul grâce à K-means et minimise la consommation d'énergie grâce à l'algorithme d'élimination gaussien par rapport aux protocoles LEACH [24] et à LEACH-C [159].

5. Le protocole Randhawa et al.

Dans [233], Randhawa et al. ont proposé deux approches de clustering une basée sur K-Means et l'autre sur une version améliorée de K-Means pour analyser leurs impacts sur la consommation d'énergie des nœuds en faisant varier le type de paramètres d'entrée. Dans la première approche (la version standard de K-Means), l'élection des CHs est faite en fonction de la distance alors que dans la deuxième, elle est faite en fonction de la distance et l'énergie restante des nœuds. Les résultats de simulations ont montré que la deuxième approche est meilleure en termes de consommation d'énergie et de durée de vie du réseau comparée à la première approche et à LEACH [24]. De même, la première approche surpasse LEACH.

6. Le protocole KM-LEACH

Dans [234], les auteurs ont proposé une version améliorée de LEACH appelée "improved K-Means cluster-based routing scheme for WSNs (KM-LEACH)" à laquelle est ajoutée l'approche de clustering K-Means. Cette approche permet d'élire les nœuds qui sont proches des centroïdes et dont l'énergie restante est supérieure à un certain seuil comme CHs. Ensuite, les mêmes procédures pour la formation des clusters et la transmission des données décrites dans LEACH [24] sont utilisées dans KM-LEACH. Les résultats de simulations ont illustré que KM-LEACH réduit la consommation d'énergie et prolonge la durée de vie des RCSFs par rapport au protocole LEACH.

7. Le protocole MRRCE

"Multi-hop Routing Reducing Consumed Energy protocol (MRRCE)" [235] est un protocole de routage hiérarchique multi-sauts basé sur la méthode K-Means et des points Steiner. Dans ce protocole, les points Steiner permettent l'élection des CHs,

ensuite les positions des CHs sont utilisées comme centroïdes initiaux dans la méthode K-Means pour former les clusters. Une fois que la formation des clusters est terminée, le processus de transmission des données est lancé. Chaque nœud envoie ses données collectées à son CH correspondant. Puis, chaque CH agrège toutes les données reçues en un seul paquet et l'envoie à la station de base par le biais des CHs qui ont plus d'énergie et qui sont proches de la station de base. Les résultats de simulations ont montré que MRRCE surpasse LEACH [24], DIRECT [236], HEED [226], EHEED [237], et TDTCEG [238] en termes de consommation d'énergie et durée de vie du réseau.

8. Le protocole de Mahboub et al.

Dans [239], un protocole de routage basé sur une hybridation de LEACH et de la méthode K-Means dans les RCSFs. La méthode K-Means est utilisée pour la formation des clusters et le protocole LEACH est utilisé au sein de chaque cluster pour l'élection des CHs. Les résultats de simulations ont montré que le protocole proposé offre de meilleures performances par rapport au protocole LEACH en termes de consommation d'énergie et de durée de vie du réseau. Cependant, l'inconvénient majeur de ce protocole est la grande consommation d'énergie lors de l'élection des CHs.

9. Le protocole C.R.D.A

"Clustering Routing based on Dijkstra Algorithm (C.R.D.A)" [240] est un protocole de routage basé sur la théorie des jeux, K-Means et Dijkstra pour réduire la consommation d'énergie dans les RCSFs. Ce protocole utilise un modèle de théorie des jeux pour trouver un emplacement optimal de la station de base, la méthode ELBOW pour définir le nombre de clusters, K-Means pour la formation des clusters et Dijkstra pour l'établissement des plus courts chemins entre chaque CH et la station de base. Dans le protocole C.R.D.A, l'élection des CHs dépend de deux critères: l'énergie résiduelle des nœuds et la distance qui les sépare du centroïde de leur propre cluster. L'algorithme Dijkstra est appliqué pour sélectionner l'ensemble des nœuds relais fiables minimisant la puissance de transmission pour acheminer des données inter et intra cluster. Les résultats de simulations ont démontré que le protocole proposé réduit la consommation d'énergie et prolonge la durée de vie du réseau par rapport au protocole LEACH.

2.3 Conclusion

Dans ce chapitre, nous avons proposé une classification des protocoles de routage conçus pour les réseaux de capteurs sans fil. Cette classification permet de classer ces protocoles en deux grandes familles: les protocoles de routage basiques et les protocoles de routage basés sur les méthodes d'optimisation. La première famille concerne les protocoles de routage conçus pour les RCSFs de petite et de moyenne taille et qui impliquent des schémas de routage déterministes alors que la deuxième famille a été conçue en particulier pour les réseaux de capteurs à grande échelle là où l'exploration de tous les chemins pour trouver le meilleur ne pourra pas se faire dans un temps raisonnable. En outre, dans cette

Table 2.1: Récapitulatif sur les protocoles basés sur le clustering

Protocole	Méthode de clustering	de	Processus de clustering	Communication (Intra/Inter)	Objectif
LEACH [24]	Probabiliste		Distribué	Mono-saut/Mono-saut	Rotation des CHs
FCH [194]	Logique Floue		Centralisé	Mono-saut/Mono-saut	↓ Énergie
T-ANT [172]	ACO		Distribué	Multi-sauts	↓ Overhead
GA-ECEP [134]	AG		Centralisé	Mono-saut/Mono-saut	↓ Énergie
PSO-C [158]	PSO		Centralisé	Mono-saut/Mono-saut	Optimiser l'énergie
Cordina et al. [221]	RN avec 4 entrées		Centralisé	Mono-saut/Mono-saut	Équilibrage de charge
CHEF [195]	Logique Floue		Distribué	Mono-saut/Mono-saut	↓ Overhead & ↓ Énergie
ACA-LEACH [174]	Probabiliste		Distribué	Mono-saut/Multi-sauts	↓ Énergie
CLIQUE [212]	AR		Distribué	Multi-sauts/Multi-sauts	↑ Durée de vie
LEACH-FL [196]	Logique Floue		Centralisé	Mono-saut/Mono-saut	↓ Énergie
LEACH-GA [136]	Probabiliste, AG		Distribué	Mono-saut/Mono-saut	↑ Durée de vie
GFCM [137]	FCM, AG		Centralisé	Mono-saut/Mono-saut	↑ Durée de vie
GABEEC [138]	AG		Centralisé	Mono-saut/Mono-saut	↑ Durée de vie
ABC-C [185]	ABC		Centralisé	Mono-saut/Mono-saut	↑ Durée de vie
LEACH-KED [230]	K-Means		Centralisé	Mono-saut/Mono-saut	↓ Énergie
ICT2TSK [197]	Logique Floue		Centralisé	Mono-saut/Mono-saut	↓ Énergie & ↑ Durée de vie
M2NGA [140]	AG à 2 niveaux		Centralisé	Multi-sauts/Multi-sauts	↑ Durée de vie
Zhang et Yifei [241]	AG		Centralisé	Mono-saut/Multi-sauts	↑ Durée de vie
SEP-FL [200]	Logique Floue		Centralisé	Mono-saut/Mono-saut	↑ Stabilité du réseau
PSO-HC [160]	PSO		Centralisé	Multi-sauts/Mono-saut	↑ Durée de vie & passage à l'échelle
KEAC [231]	K-Means		Centralisé	Mono-saut/Mono-saut	↓ Énergie & ↓ Overhead
OZEEP [144]	AG, LF		Centralisé	Mono-saut/Multi-sauts	Équilibrage de charge & Tolérance aux pannes
Behrang et al. [149]	K-Means,AG		Centralisé	Mono-saut/Mono-saut	Équilibrer la distribution de l'énergie
CBMR [125]	Déterministe		Centralisé	Mono-saut/Multi-sauts	Fiabilité du réseau & ↓ Énergie
DFLC [202]	Logique Floue		Distribué	Multi-sauts/Mono-saut	Tolérance aux pannes & Équilibrage de charge & Passage à l'échelle
MLRC [117]	Min-max normalisation		Distribué	Mono-saut/Multi-sauts	Réduire le nombre de transmissions & ↑ Durée de vie
GADA-LEACH [150]	AG		Centralisé	Mono-saut/Multi-sauts	↑ Durée de vie
ABC-SD [191]	ABC		Centralisé	Multi-sauts/Multi-sauts	↓ Énergie & Passage à l'échelle
MRRCE [235]	Steiner points, K-Means		Centralisé	Mono-saut/Multi-sauts	↓ Énergie
FBUC [205]	Logique Floue		Distribué	Mono-saut/Mono-saut	↓ Délai & ↑ Durée de vie
PSO-ECHS [167]	LP, PSO		Centralisé	Mono-saut/Mono-saut	↓ Énergie
EEDCF [206]	Logique Floue		Distribué	Multi-sauts/Mono-saut	↓ Énergie
Mahboub et al. [239]	K-Means, probabiliste		Semi-distribué	Mono-saut/Mono-saut	↓ Énergie
KGA [152]	AG, K-Means		Centralisé	Mono-saut/Multi-sauts	↑ Durée de vie
C.R.D.A [240]	ELBOW method, K-Means		Centralisé	Mono-saut/ Multi-sauts	↓ Énergie
O-LEACH [154]	Probabiliste, AG		Distribué	Mono-saut/Multi-sauts	↑ Durée de vie

deuxième famille nous avons mis l'accent sur les protocoles qui impliquent le paradigme clustering dans la hiérarchisation des réseaux en vue d'augmenter leurs performances.

Dans le chapitre suivant, nous proposons une étude sur les méthodes les plus répandues permettant la détermination du nombre optimal de clusters. Cette étude sera suivie d'une évaluation de leurs performances dans les RCSFs.

Chapitre 3

Apport du clustering sur le routage dans les RCSFs

Chapter 3

Apport du clustering sur le routage dans les RCSFs

3.1 Introduction

Un RCSF est un ensemble de capteurs ayant des ressources limitées en termes de puissance, de batterie, de calcul et de communication. Il est généralement constitué d'un grand nombre de capteurs contrôlés par une ou plusieurs stations de base. Les capteurs peuvent détecter, traiter et ensuite transmettre les données à la station de base à laquelle ils sont attachés. Les RCSFs ont été impliqués dans plusieurs applications telles que la santé, l'industrie, les systèmes militaires, civils et de transport [242–245].

Les défis des diverses applications dans les RCSFs, comme cela a été discuté dans [246, 247] demandent aux capteurs de prendre en charge non seulement des communications économes en énergie, mais également des supports sensibles aux délais. Ces défis peuvent inclure également la tolérance aux pannes, la sécurité, la gestion du passage à l'échelle, le routage, la qualité de service, la mobilité et l'énergie. Néanmoins, le principal défi des RCSFs est de réduire la consommation d'énergie et de prolonger la durée de vie des réseaux, car les capteurs sont généralement équipés de batteries à faible puissance, non rechargeables et leur remplacement est difficile voire impossible surtout dans les endroits hostiles. À cet effet, l'efficacité énergétique est considérée comme le principal défi lors de la conception d'un protocole de routage pour les RCSFs. Dans cette optique, différentes approches de routage visant la conservation de l'énergie ont été proposées pour préserver l'énergie dans les RCSFs: des solutions softwares et hardwares [248–250].

Les protocoles de routage conçus pour les RCSFs peuvent être divisés en trois classes principales: les protocoles de routage à plat, les protocoles de routage hiérarchiques et les protocoles de routage basés sur la localisation. Dans le routage à plat, tous les nœuds ont le même statut et sa mise en place est simple mais il présente quelques inconvénients, tels que le manque de gestion des nœuds qui rend difficile l'optimisation des ressources du réseau et la lenteur de la réponse aux changements dynamiques du réseau. Ces inconvénients ont rendu ce type de protocoles non convenables aux réseaux de capteurs à grande échelle. Dans les protocoles de routage basés sur la localisation, les capteurs doivent réserver les informations de localisation de leurs voisins. Par conséquent, ce processus peut entraîner une surcharge de routage. Les protocoles de routage hiérarchiques divisent les capteurs en plusieurs clusters, où chacun cluster est composé de plusieurs nœuds membres dont

un seul clusterhead qui les supervise [104]. Les CHs sont sélectionnés parmi les capteurs de chaque cluster aléatoirement ou en fonction de certains critères. Les membres d'un cluster recueillent des informations et les envoient à leur CH correspondant qui à son tour agrège ces informations en un seul paquet pour éviter toute redondance de transmissions de données et le transmet directement ou via un schéma de routage multi-sauts à la station de base.

Par ailleurs, l'approche basée sur le clustering permet une utilisation efficace des ressources énergétiques [105]. Cependant, l'un des problèmes épineux dans ce paradigme est de savoir comment déterminer le nombre optimal de clusters afin d'améliorer les performances de routage dans les RCSFs ?

La qualité du processus de clustering repose sur le bon choix du nombre de clusters, la manière de formation des clusters et sur l'élection des CHs. Ces trois facteurs ont un impact sur les performances des RCSFs, en termes du passage à l'échelle, d'efficacité énergétique, de réduction des collisions, de durée de vie des réseaux et de latence. De plus, si les clusters ne sont pas construits de manière optimale, l'énergie totale consommée par les RCSFs durant une période (round) sera augmentée de façon exponentielle lorsque le nombre de clusters créés est supérieur ou inférieur au nombre de clusters optimal [251].

Dans ce chapitre, nous sommes intéressés à l'estimation du nombre de clusters en comparant trois méthodes qui sont considérées parmi les plus répandues dans la littérature (la méthode Silhouette, la méthode Elbow et la méthode "Rule of Thumb") afin de trouver la méthode la plus efficace en termes de consommation d'énergie, de durée de vie du réseau et de nombre de paquets envoyés à la station de base dans les RCSFs. Ces trois méthodes de clustering ont fourni de meilleures performances dans plusieurs domaines. Les méthodes Silhouette et Elbow ont été largement utilisées et elles sont populaires dans plusieurs domaines. Ces méthodes visent à trouver la valeur optimale de k en recherchant la meilleure manière de grouper les capteurs les plus proches pour plusieurs valeurs de k . De plus, la méthode "Rule of Thumb" utilise la racine carrée de la moitié du nombre de nœuds pour déterminer la valeur k . À notre connaissance, la méthode "Rule of Thumb" n'a pas été utilisée comme méthode de clustering pour les RCSFs. De plus, le nombre de clusters généré par les méthodes Silhouette et Elbow est en relation avec la topologie du réseau et indépendamment de la taille du réseau, tandis que la méthode "Rule of Thumb" génère un nombre de clusters en fonction de la taille du réseau.

La performance des méthodes de clustering appliquées à de nombreux domaines repose sur la détermination d'un nombre minimal de clusters. Toutefois, dans les RCSFs, la performance d'une telle méthode est liée à la consommation d'énergie et à la durée de vie de ces réseaux, c'est-à-dire la méthode qui permet de minimiser la consommation d'énergie et prolonger la durée de vie des réseaux est considérée comme la meilleure méthode. Dans ce contexte, nous proposons d'appliquer trois méthodes de classification différentes qui génèrent différents nombre de clusters dans RCSFs [18].

Dans ce travail, nous étudions certaines méthodes qui sont considérées parmi les plus répandues dans la littérature pour déterminer le nombre de clusters: la méthode "Rule of Thumb" [19], la méthode Silhouette [21, 22] et la méthode Elbow [252]. Après avoir déterminé le nombre de clusters par chacune d'elles, nous appliquons une approche distribuée basée sur l'approche de clustering K-Means [23] pour former les clusters, puis nous évaluons les performances de chacune d'elles en termes de nombre de paquets envoyés à la station de base, de l'énergie consommée et du nombre de nœuds vivants. De

plus, nous comparons les résultats obtenus avec les protocoles de routage LEACH [24], Imp_LEACH [25] et MODLEACH [26].

Dans ce qui suit, nous donnons un aperçu général sur l'ensemble des techniques utilisées dans ce chapitre. Ensuite, nous présentons tout d'abord les travaux connexes pertinents dans ce contexte. Puis, nous présentons comment mettre en œuvre des schémas de routage basés sur ces méthodes de clustering tout en analysant la complexité de la sélection des CHs. Finalement, nous évaluons les performances de chacune de ces méthodes et nous les comparons entre elles.

3.2 Préliminaires

Dans cette section, nous présentons brièvement l'approche de clustering K-Means et trois méthodes qui sont considérées comme les plus répandues pour déterminer le nombre optimal de clusters: la méthode "Rule of Thumb", la méthode Silhouette et la méthode Elbow.

3.2.1 L'approche K-Means

K-Means [23] est l'un des algorithmes d'apprentissage non supervisé le plus simple permettant de résoudre le problème de clustering. La procédure utilisée par K-Means suit une méthode simple pour classer un ensemble de données en fonction d'un nombre prédéfini de clusters (supposons que K est le nombre de clusters) qui est fixé à l'avance. L'idée principale est de définir les K centroïdes, un pour chaque cluster. Ces centroïdes doivent être placés le plus loin possible les uns des autres. L'étape suivante consiste à prendre chaque point appartenant à un ensemble de données et l'associer au centroïde le plus proche. Lorsqu'aucun point n'est en attente, la première étape est terminée et des clusters sont formés. À ce stade, k nouveaux centroïdes sont recalculés comme barycentres des clusters résultants de l'étape précédente. Après avoir calculé les k nouveaux centroïdes, une nouvelle liaison doit être effectuée entre les mêmes points de jeu de données et le nouveau barycentre le plus proche. Ainsi, une boucle est générée. À la suite de cette boucle, nous pouvons remarquer que les k centroïdes changent d'emplacement à chaque itération jusqu'à ce qu'aucune autre modification ne soit apportée, c'est-à-dire quand les centroïdes ne bougent plus. Enfin, cette technique vise à minimiser une fonction objectif représentée par l'équation 3.1.

$$\text{Min } f(x) = \sum_{r=1}^K \sum_{(x_i \in C_r)} \|x_i - g_r\|^2 \quad (3.1)$$

où $\|x_i - g_r\|^2$ est la distance Euclidienne entre l'individu x_i et le centre du cluster g_r , C_r est la classe numéro r , x_i est un individu dans un cluster, et g_r est le centre de classe C_r .

K-Means est l'approche de clustering la plus simple et la plus adaptée aux RCSFs pour les structurer en clusters et trouver les bons cluster-heads pour acheminer les données vers la station de base.

L'exécution de la méthode K-Means suit les étapes suivantes:

1. Placez les K points dans l'espace. Ces points représentent les centroïdes initiaux des clusters .
2. Associez chaque point au cluster ayant le centroïde le plus proche.
3. Lorsque tous les points sont affectés, recalculez les positions des K centroïdes.
4. Répétez les étapes 2) et 3) jusqu'à ce que les centroïdes ne changent plus. Ceci produit une séparation des points en clusters à partir desquels la métrique à minimiser peut être calculée.

3.2.2 Méthode "Rule of Thumb"

La méthode "Rule of Thumb" [19] est l'une des méthodes les plus simples pour déterminer le nombre de clusters. Elle peut s'appliquer à tout type de jeu de données. Cette méthode est composée des étapes suivantes:

1. Trouvez la valeur de K qui représente le nombre de clusters à générer en utilisant l'équation 3.2.

$$K = \sqrt{N/2} \quad (3.2)$$

Où N est le nombre de points de jeu de données

2. Appliquez la méthode K-Means pour générer les clusters.

3.2.3 Méthode Silhouette

Le concept de la largeur de Silhouette (Silhouette width) [21, 22] traite la différence entre la cohésion du cluster et la séparation des autres. En particulier, il fournit une moyenne quantitative pour mesurer la position de chaque point dans son cluster par rapport aux autres. La moyenne Silhouette de chaque point est une mesure de la similitude de ce point avec les points de son propre cluster par rapport aux points des autres clusters. Le coefficient de Silhouette d'un point de données est défini par l'équation 3.3.

$$S(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \quad (3.3)$$

Où $a(i)$ est la distance moyenne entre le point i et les autres points du même cluster. Soit $b(i, k)$ la distance moyenne du point i aux points du cluster k alors $b(i)$ est le minimum de $b(i, k)$ sur tous les clusters auxquels le point i n'appartient pas. La valeur de $S(i)$ ne doit pas dépasser 1, et si $S(i)$ est proche de 1, cela signifie que le point i se situe bien dans son propre cluster.

La méthode Silhouette est composée des étapes suivantes:

1. Exécutez K-Means en faisant varier K (le nombre de clusters) de 1 à max,
2. Pour chaque K , calculez la silhouette moyenne des observations,
3. Tracez la courbe de "avg.sil" en fonction du nombre de clusters K ,
4. L'emplacement du maximum est considéré comme le nombre de clusters approprié.

3.2.4 Méthode Elbow

Elbow [252] est la méthode la plus ancienne pour déterminer le nombre approprié de clusters dans un ensemble de données. C'est une méthode visuelle dont l'idée est de commencer par $K = 2$ et de l'augmenter progressivement (en ajoutant 1 après chaque itération), puis déterminer les clusters et calculer le coût de leur formation. À une certaine valeur de K , le coût diminue considérablement et ensuite, il atteint un seuil lorsqu'on l'augmente davantage. Il s'agit donc de la meilleure valeur de K (nombre de clusters). Les étapes de la méthode Elbow sont les suivantes:

1. Exécutez l'algorithme K-Means pour différentes valeurs de K . Le nombre de k peut prendre la valeur de 1 au nombre d'instances,
2. Pour chaque K , calculez la somme totale du carré au sein d'un cluster "Total within-cluster sum of square (WSS)",
3. Tracez la courbe de WSS en fonction de chaque valeur de K ,
4. Le point où le graphe marginal va chuter de façon remarquable. Ce point représente le nombre de clusters K recherché.

La Figure 3.1 illustre le meilleur nombre de clusters (le cercle rouge est le point du coude (Elbow)).

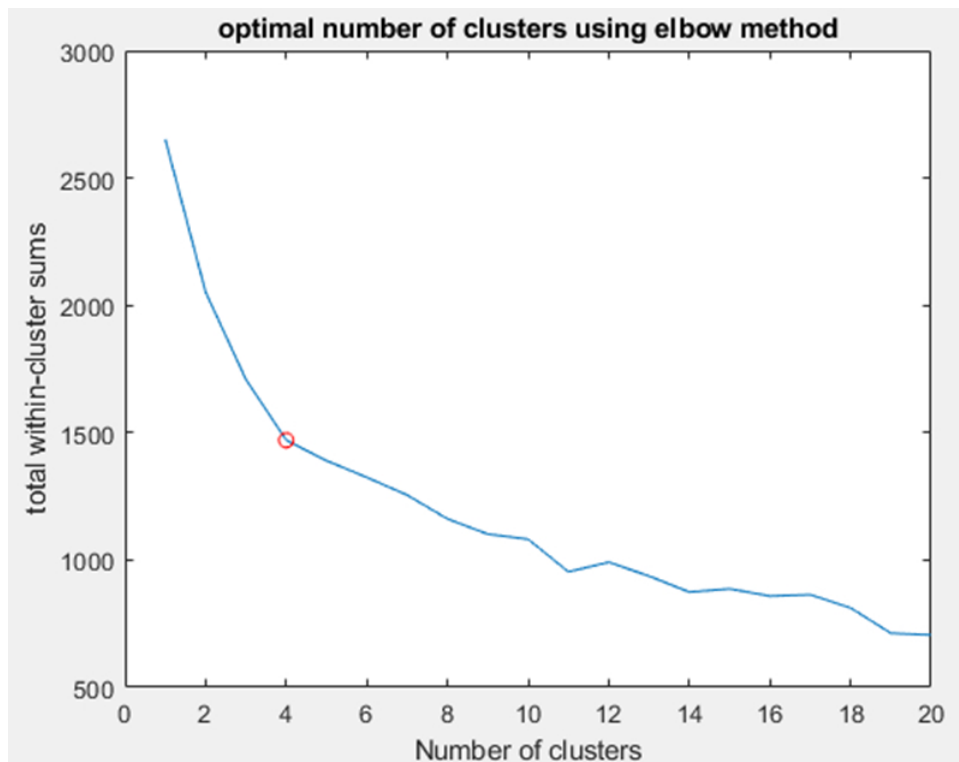


Figure 3.1: Identification du point Elbow

3.3 Travaux connexes

Les protocoles de routage efficaces en énergie basés sur le clustering ont attiré beaucoup d'attention dans les RCSFs grâce à leur utilisation efficace des ressources énergétiques. Dans ces protocoles, les nœuds du réseau sont divisés en petits groupes appelés clusters. L'un des nœuds dans chaque cluster est choisi comme CH et les autres nœuds sont appelés nœuds membres. Les nœuds membres envoient leurs données détectées à leur CH correspondant. Le CH effectue un certain type d'agrégation de données et génère un seul paquet de données, puis le transmet à la station de base. Néanmoins, l'un des problèmes importants dans le paradigme de clustering est de savoir comment déterminer le nombre optimal de clusters afin d'améliorer les performances de routage dans les RCSFs.

Dans cette section, nous avons mis l'accent sur certains protocoles de routage basés sur le clustering en utilisant une méthode particulière pour déterminer le nombre de clusters. Dans [253], les auteurs associent la méthode Elbow et la méthode K-Means au protocole LEACH [24, 159] pour concevoir un protocole économe en énergie. La méthode Elbow est utilisée pour déterminer le nombre de clusters, K-Means pour générer les clusters et le protocole LEACH pour l'élection des CHs et la transmission de données. Cette combinaison a permis d'améliorer la durée de vie des réseaux de capteurs. Toutefois, la durée nécessaire pour structurer le réseau en clusters était très longue.

Dans [232], les auteurs ont proposé un protocole de routage basé sur le clustering dans lequel ils utilisent une version améliorée de la méthode K-Means pour former des clusters et un algorithme d'élimination gaussien pour le choix des CHs. Le protocole de routage proposé a fourni de bons résultats mais sa durée d'exécution est plus longue.

Dans [239], les auteurs ont proposé un protocole de routage appelé "K-medoids hybrid (KMV-LEACH)" dans lequel la technique K-medoids est utilisée pour générer des clusters uniformes et le protocole LEACH pour sélectionner les CHs. Dans ce travail, les auteurs comparent les performances de trois méthodes pour déterminer le nombre optimal de clusters (Gapstatistic, CalinskiHarabasz et Silhouette) et proposent un schéma de routage basé sur la méthode Silhouette. Le protocole proposé donne de bons résultats, mais il génère un overhead important.

Dans [254], un protocole de routage est proposé pour réduire la consommation d'énergie et prolonger la durée de vie des réseaux. Les auteurs ont utilisé l'algorithme K-medoids pour calculer les medoids optimaux entre les nœuds. Ensuite, ils ont sélectionné les CHs appropriées en équilibrant la charge du réseau entre les clusters. Ainsi, l'efficacité énergétique est améliorée et la durée de vie du réseau est prolongée de manière efficace. Les résultats de simulations ont montré que l'algorithme proposé surpasse d'autres algorithmes en termes de consommation d'énergie et de durée de vie du réseau.

Dans [25], les auteurs ont proposé un nouveau schéma de routage hiérarchique basé sur le clustering et sur le protocole de routage LEACH afin de maximiser la durée de vie des réseaux. La solution proposée vise à minimiser la consommation d'énergie lors de la transmission de données à la station de base (BS) et du nombre de nœuds morts éloignés de la station de base. Cette amélioration évite la mort des nœuds très éloignés de la station de base et donne plus de chance aux nœuds situés à la moitié de la topologie pour devenir des CHs. Les résultats de simulations ont montré que cette solution garantit une faible consommation d'énergie.

Dans [255], les auteurs proposent un protocole de routage multi-sauts et orienté basé

sur le clustering pour les RCSFs, appelé DMCR. DMCR choisit le chemin le plus fiable pour la transmission de données et qui implique le moins de noeuds intermédiaires. Le processus de routage est effectué à l'aide de dispositifs peu coûteux, tels que des antennes "Angle of Arrival (AoA)". Ce schéma de routage réduit le nombre de noeuds participant à la transmission de données à la station de base et implique les noeuds qui ont plus d'énergie et les plus proches de la station de base dans chaque cluster pour relayer les données agrégées. Le protocole proposé a fourni de meilleurs résultats en termes de durée de vie du réseau et de livraison de paquets de données par rapport à d'autres protocoles.

Compte tenu de ces inconvénients, dans ce qui suit nous allons évaluer et comparer les performances de trois méthodes plus connus permettant de déterminer le nombre optimal de clusters dans les RCSFs pour tirer parti de leurs avantages.

3.4 Schémas de routage proposés

Avant de présenter en détails les schémas de routage associés à chacune des méthodes de clustering décrites dans la section 3.2, nous supposons les hypothèses suivantes:

- Les capteurs sont déployés de manière aléatoire dans la zone d'intérêt et ils ne sont pas mobiles.
- La station de base est située au centre de la zone de déploiement du réseau,
- Les capteurs peuvent ajuster la puissance de transmission en fonction de la distance qui les sépare du noeud destinataire.
- Au début, tous les capteurs ont la même quantité d'énergie,
- L'emplacement et l'ID de chaque capteur sont connus par la station de base,
- Chaque capteur connaît sa propre position.

Dans cette section, nous présentons comment les schémas de routage basés sur les techniques citées dans la section 3.2 sont mis en œuvre. Chaque schéma de routage se déroule en deux phases: la phase de configuration et la phase de transmission de données comme illustré par la Figure 3.2.

3.4.1 Phase de configuration

Cette phase comprend trois sous-étapes: la détermination du nombre de clusters, la formation des clusters et la phase d'élection des CHs. La première et la deuxième sous-étapes ne sont exécutées que durant la première période, alors que la troisième sous-étape est exécutée après chaque période.

a) Détermination du nombre de clusters

Tout d'abord, nous déterminons le nombre approprié de clusters générés par chaque approche (Silhouette, Elbow et "Rule of Thumb"). Nous proposons deux scénarios pour évaluer les performances de chacune de ces trois approches. Dans le premier scénario,

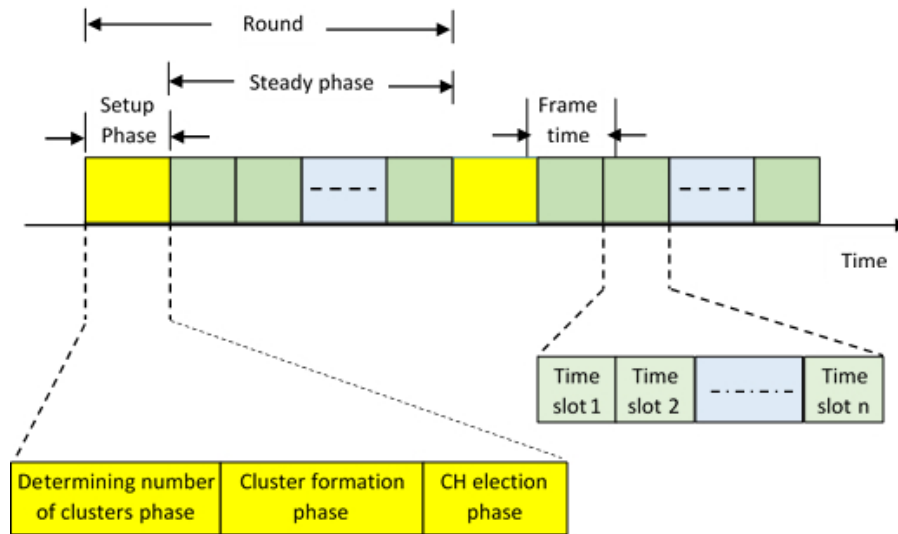


Figure 3.2: Phases des schémas de routage

nous avons choisi un réseau contenant 200 nœuds, tandis que dans le deuxième scénario, nous avons choisi un réseau contenant 500 nœuds, afin d'illustrer l'impact du clustering sur les performances de chaque méthode lors du passage à l'échelle.

Après avoir exécuté les simulations associées à chaque scénario, chacune des méthodes a fourni les résultats suivants, comme indiqué dans le tableau 3.1.

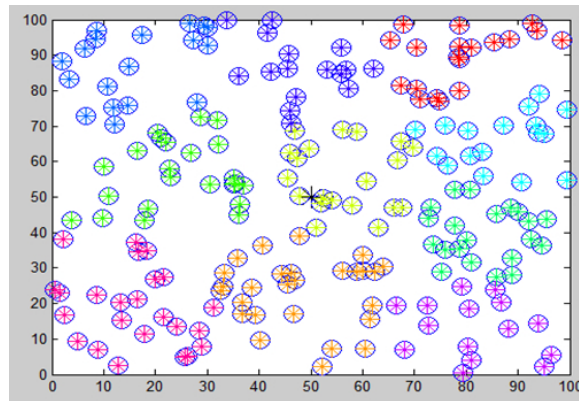
Table 3.1: Détermination du nombre de clusters par chaque méthode

Taille du réseau	Nombre de clusters		
	Rule of Thumb	Silhouette	Elbow
200 noeuds	10	4	5
500 noeuds	16	10	6
2000 noeuds	32	4	6

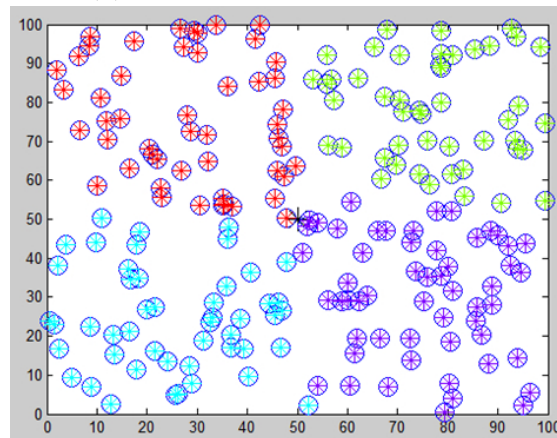
b) Formation des clusters

Après avoir déterminé le nombre de clusters par chaque méthode, le processus de clustering (formation de clusters) utilise la méthode K-Means décrite dans la section 3.2. En outre, pour générer des clusters plus ou moins équilibrés en termes de nombre de nœuds par cluster, nous avons déterminé les centroïdes initiaux en fonction de la densité des nœuds dans la zone de déploiement au lieu de les choisir de manière aléatoire.

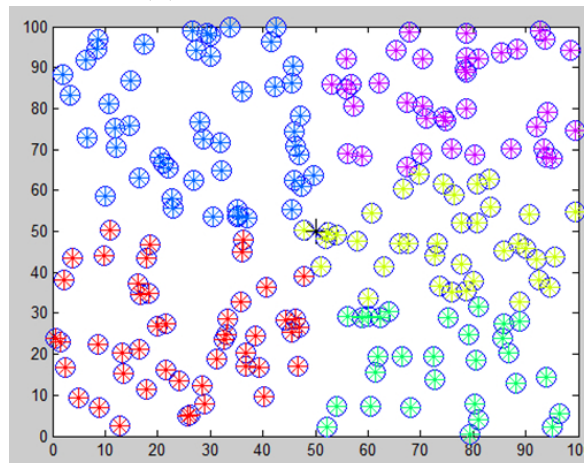
Les Figures 3.3 et 3.4 montrent respectivement comment les clusters sont formés à l'aide de la méthode K-Means par chacune des trois approches de clustering avec un réseau de 200 nœuds et un réseau de 500 nœuds.



(a) Rule of Thumb: 10 Clusters



(b) Silhouette: 4 Clusters

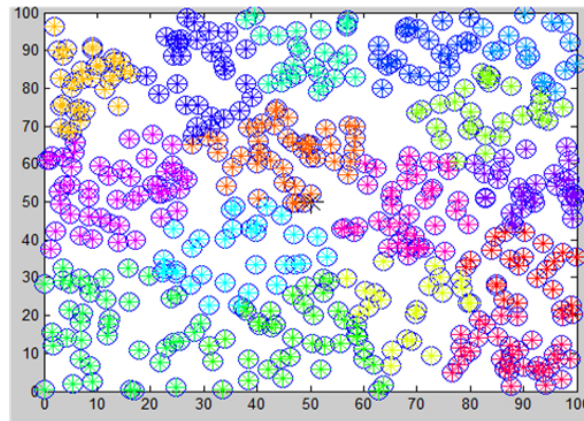


(c) Elbow: 5 Clusters

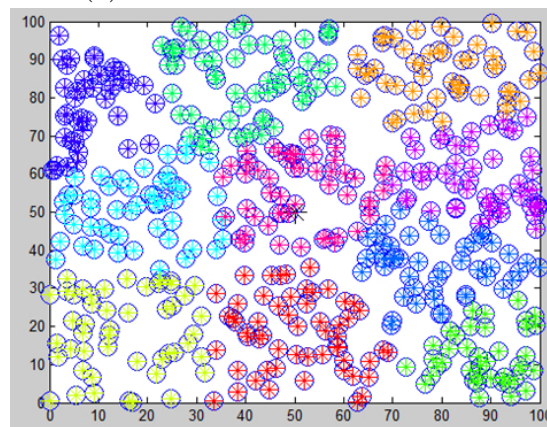
Figure 3.3: Nombre de clusters créés dans un réseau contenant 200 noeuds

c) Election des clusterheads

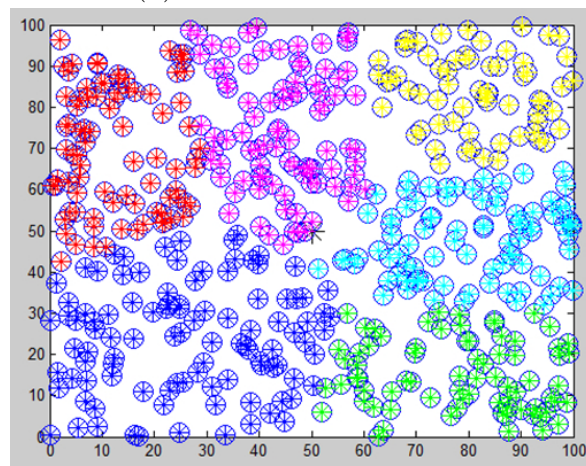
Puisque les CHs sont chargés de plusieurs tâches, telles que la responsabilité des nœuds membres du cluster, l'agrégation des données reçues et leur transmission à la station de base, directement ou via une transmission multi-sauts, ils peuvent donc rapidement épuiser leurs batteries. De plus, ces nœuds sont limités en énergie et leurs batteries ne



(a) Rule of Thumb: 16 Clusters



(b) Silhouette: 10 Clusters



(c) Elbow: 6 Clusters

Figure 3.4: Nombre de clusters créés dans un réseau contenant 500 noeuds

sont généralement pas rechargeables et leur remplacement est difficile voire impossible dans la plupart des cas, en particulier lorsqu'ils sont déployés dans des environnements hostiles.

Par ailleurs, l'épuisement de l'énergie des capteurs entraîne la cessation de fonction de ces derniers ce qui peut causer l'apparition de zones non couvertes et peut aussi affecter

la fiabilité de livraison de données à la station de base. De ce fait, nous avons évité que les CHs soient élus une seule fois durant toute la durée de vie du réseau et nous avons proposé que leur élection sera faite après chaque période pour une meilleure gestion de la consommation de l'énergie entre les noeuds. Le processus d'élection des CHs se déroule de la manière suivante: au début de chaque période, chaque capteur calcule son poids en fonction de son énergie restante et de la distance qui le sépare du centroïde correspondant. Ce poids est calculé comme le montre l'équation 3.4.

$$Cost(Node_i) = \alpha * Energy + \frac{\beta}{dist(Node_i, Centroid)} \quad (3.4)$$

$$\alpha + \beta = 1 \quad \alpha \in]0.75, 0.95[, \quad \beta \in]0.05, 0.25[$$

De plus, le poids de chaque nœud est calculé périodiquement afin d'illustrer l'aptitude d'un nœud à jouer le rôle de CH du cluster. Cette stratégie permet de prolonger la durée de vie des capteurs. Dans chaque cluster, le nœud ayant le poids le plus élevé parmi les membres du cluster devient CH durant la période courante.

3.4.2 Phase de transmission de données

Dans les RCSFs, les nœuds basculent périodiquement entre deux modes: le mode veille et le mode actif afin d'économiser de l'énergie. Ainsi, pour effectuer le basculement entre les deux modes, un mécanisme TDMA (un mécanisme d'accès multiple par division temporelle) est utilisé comme dans LEACH où au cours duquel le capteur se met en mode actif pour une durée time-slot pour envoyer les données collectées à son CH correspondant et après l'expiration de ce time-slot le capteur retourne au mode veille. Par ailleurs, dans ce mécanisme, un message beacon est envoyé par le CH aux nœuds membres de son cluster pour synchroniser entre eux. De ce fait, chaque nœud transmet ses données collectées lors de son time-slot pour éviter les interférences avec les autres membres du même cluster comme c'est présenté dans la Figure 3.2. Les membres du cluster ne transmettent pas leurs données collectées directement à la station de base, mais à leur CH correspondant. Ce dernier agrège les données collectées par les nœuds membres dans un seul paquet de données et l'envoie directement à la station de base, comme indiqué dans la Figure 3.5. Cette opération d'agrégation de données permet d'éviter la transmission de données redondantes et minimiser le nombre de données envoyées à la station de base, ce qui permet de préserver de l'énergie et par conséquent prolonger la longévité du réseau.

3.4.3 Complexité de l'élection des CHs

La faisabilité du processus de clustering peut être déterminée par la complexité du processus d'élection des CHs. Dans cette section, nous analysons cette complexité.

Dans le modèle proposé, le RCSF est composé d'un grand nombre de nœuds déployés de manière aléatoire dans une zone d'intérêt. Tous ces nœuds sont regroupés dans des clusters. Dans chaque cluster, un seul nœud agira comme CH et les autres nœuds restants agiront comme nœuds membres. Les membres du cluster transmettent les données collectées à leur CH correspondant, qui les agrège à son tour en un seul paquet de données et l'envoie directement à la station de base. Ce schéma de routage permet d'économiser

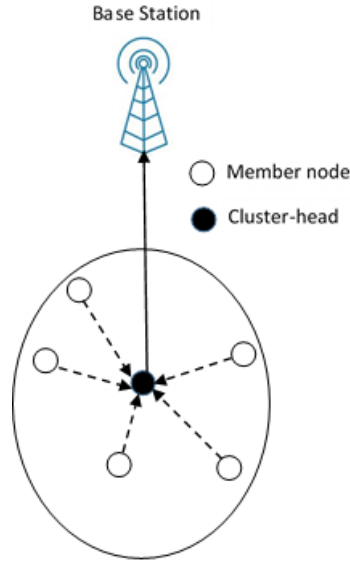


Figure 3.5: Communications inter-cluster et intra-cluster

plus d'énergie et d'améliorer la durée de vie du réseau. Le processus de clustering est divisé en deux phases: la phase de configuration et la phase de transmission de données.

a) Complexité temporelle

La phase de configuration est exécutée dans un temps très court par rapport à la phase de transmission de données. Par conséquent, la complexité temporelle de la phase de configuration et en particulier la complexité temporelle de l'opération d'élection des CHs peut être négligée. Ainsi, nous évaluons la complexité en termes de messages échangés lors de la phase d'élection des CHs.

b) Complexité en termes de messages échangés

Dans cette contribution, nous considérons un RCSF qui est composé de n nœuds et qu'il est divisé en k clusters. Soit nb_{Cl} le nombre moyen de membres par cluster, comme illustré par l'équation 3.5.

$$nb_{Cl} \approx \frac{n}{k} \quad (3.5)$$

Dans le processus de sélection des CHs, chaque nœud diffuse un message beacon contenant son poids aux membres de son propre cluster et chaque membre du cluster reçoit ce message.

1. Diffusion des messages beacons

Le nombre de messages beacons envoyés nb_msg_{sent} dans un cluster est égal au nombre de nœuds dans ce cluster :

$$nb_msg_{sent} = nb_{Cl}$$

Par conséquent, le nombre total de messages beacons transmis durant le processus d'élection des CHs est égal à:

$$nb_tot_sent = k \times \underbrace{nb_msg_sent}_n$$

D'où, la complexité en termes de nombre de messages beacons envoyés par tous les nœuds du réseau au cours du processus d'élection des CHs est de l'ordre de $\theta(n)$.

Soit m_{bits} la longueur du message beacon et E_{Tx} l'énergie nécessaire pour diffuser un message beacon, l'énergie nécessaire pour diffuser tous les messages beacons dans le réseau pendant le processus d'élection des CHs se calcule comme suit :

$$E_{Tx} = m_{bits} \times E_{1Tx}$$

Où E_{1Tx} représente l'énergie consommée pour envoyer un seul bit (1 bit).

$$E_{nTx} = n \times E_{Tx}$$

$$E_{nTx} = n \times \underbrace{m_{bits} \times E_{1Tx}}_{E_{Tx}} \quad (3.6)$$

2. Réception des messages beacons

Le nombre de messages beacons reçus $nb_msg_received$ par chaque membre d'un cluster est égal à $(nb_{Cl} - 1)$.

Il en résulte que dans un cluster le nombre de messages beacons reçus $nb_Cl_received$ par tous les membres du cluster est calculé conformément à l'équation 3.7.

$$nb_Cl_received = nb_{Cl} \times (nb_{Cl} - 1) \quad (3.7)$$

Par conséquent, le nombre total de messages beacons reçus $nb_tot_received$ par tous les nœuds du réseau pendant le processus d'élection des CHs est:

$$nb_tot_received = k \times nb_Cl_received$$

$$nb_tot_received = k \times nb_{Cl} \times (nb_{Cl} - 1)$$

$$nb_tot_received = \underbrace{k \times nb_{Cl}}_n \times (nb_{Cl} - 1)$$

D'où, la complexité en termes de nombre de messages beacons reçus par tous les nœuds du réseau durant le processus d'élection des CHs est de $\theta(n\Delta)$ où Δ représente le nombre moyen de nœuds par cluster.

De la même manière, soit m_{bits} la longueur du message beacon reçu et E_{Rx} est l'énergie nécessaire pour recevoir un message beacon comme illustré par l'équation 3.8, l'énergie nécessaire pour recevoir tous les messages beacons dans réseau au cours du processus d'élection des CHs est donné par l'équation 3.10.

$$E_{Rx} = m_{bits} \times E_{1Rx} \quad (3.8)$$

où E_{1Rx} représente l'énergie dissipée pour recevoir un bit (1 bit).

$$E_{nRx} = nb_tot_{received} \times E_{Rx}$$

$$E_{nRx} = \underbrace{k \times nb_{Cl}}_n \times (nb_{Cl} - 1) \times E_{Rx} \quad (3.9)$$

$$E_{nRx} = \underbrace{k \times nb_{Cl}}_n \times (nb_{Cl} - 1) \times \underbrace{m_{bits} \times E_{1Rx}}_{E_{Rx}} \quad (3.10)$$

Puisque l'énergie requise pour envoyer un bit est supérieure à celle requise pour le recevoir et que le nombre de messages beacons reçus est supérieur au nombre de beacons envoyés par tous les nœuds du réseau, nous pouvons supposer que la complexité des messages reçus est la même que celle des messages envoyés en terme d'énergie consommée. Nous pouvons donc en conclure que la complexité en termes de messages échangés est $\theta(n)$ pour l'élection des CHs, où n représente la taille du réseau .i.e. nombre de nœuds du réseau.

3.5 Evaluation des performances

Dans cette section, nous évaluons des schémas de routage associés à chacune des méthodes permettant de déterminer le meilleur nombre de clusters (les méthodes décrites dans la section 3.2) et nous comparons leurs performances en termes de consommation d'énergie, de durée de vie du réseau et de nombre de paquets envoyés à la station de base. Nous commençons tout d'abord par déterminer le nombre de clusters générés par chaque méthode avec trois scénarios différents: réseau ayant 200 nœuds, réseau ayant 500 nœuds et réseau ayant 2 000 nœuds.

3.5.1 Environnement de simulations

Toutes les simulations ont été réalisées en utilisant le simulateur MATLAB. Nous fournissons les résultats obtenus par les différentes méthodes de clustering (Silhouette, Elbow, Rule of Thumb, LEACH, MODLEACH, Imp_LEACH), puis nous interprétons ces résultats. Pour ce faire, nous avons utilisé trois scénarios (réseaux de 200, 500 et 2000 nœuds) dans lesquels les nœuds sont déployés aléatoirement dans une zone de $100m \times 100m$. La station de base est située au centre de la zone, c'est-à-dire aux coordonnées (50,50). Chaque nœud transmet un paquet de données par round à son CH correspondant durant le time-slot qui lui est accordé. Chaque CH agrège les données collectées par ses

membres dans un seul paquet de données et l'envoi directement à la station de base. Le tableau 3.2 récapitule les paramètres de simulation utilisés pour évaluer les performances de chaque schéma de routage associé à chaque méthode de clustering (Silhouette, Elbow, Rule of Thumb, LEACH, MODLEACH, Imp_LEACH).

Table 3.2: Paramètres de simulation des différentes méthodes de clustering

Paramètres	Valeurs
Zone de déploiement	(0,0)x(100,100)
Position de la station de base	(50,50)
ε_{elec}	50 nJ/bit
ε_{friss_amp}	10pJ/bit/m ²
$\varepsilon_{two_ray_amp}$	0.0013pJ/bit/m ⁴
d_0	87 m
Énergie initiale par chaque noeud	0.2 J
Nombre de noeuds	200, 500, 2000
P	(0.05) (5% Pour LEACH)

3.5.2 Résultats de simulations

Dans cette sous-section, nous présentons les résultats de simulations obtenus avec les trois scénarios (réseau avec 200 nœuds, réseau avec 500 nœuds et réseau avec 2 000 nœuds) et nous discutons les performances des méthodes de clustering (Silhouette, Elbow et Rule of Thumb, et celles de LEACH, Imp LEACH et MODLEACH). Nous supposons que dans LEACH, le nombre de clusters représente 5% du nombre total de nœuds déployés. Par exemple, dans le réseau dont le nombre de nœuds est 200, le nombre de clusters est égal à 10, tandis que dans celui qui contient 500 nœuds, le nombre de clusters est égal à 25.

Pour évaluer la consommation d'énergie, on a utilisé les mêmes paramètres d'énergie et le même modèle radio que ceux utilisés dans [24]. Dans le modèle radio utilisé, nous supposons que la consommation d'énergie est principalement divisée en deux parties: les réceptions et les transmissions. La consommation d'énergie pour une opération de transmission nécessite une énergie supplémentaire pour amplifier le signal en fonction de la distance qui sépare les deux noeuds communicants. Ainsi, pour transmettre un message de k bits d'une distance d , la radio dépense une quantité d'énergie comme décrit par l'équation 3.11, où ε_{elec} est l'énergie consommée par les composants de la radio, ε_{fs} et ε_{amp} pour amplifier la puissance du signal. La quantité d'énergie consommée par une opération de réception est exprimée par l'équation 3.12.

$$E_{Tx}(k, d) = \begin{cases} k * E_{elec} + \varepsilon_{fs} * d^2, & \text{if } d < d_0 \\ k * E_{elec} + \varepsilon_{amp} * d^4, & \text{if } d \geq d_0 \end{cases} \quad (3.11)$$

$$E_{Rx}(k) = k * E_{elec} \quad (3.12)$$

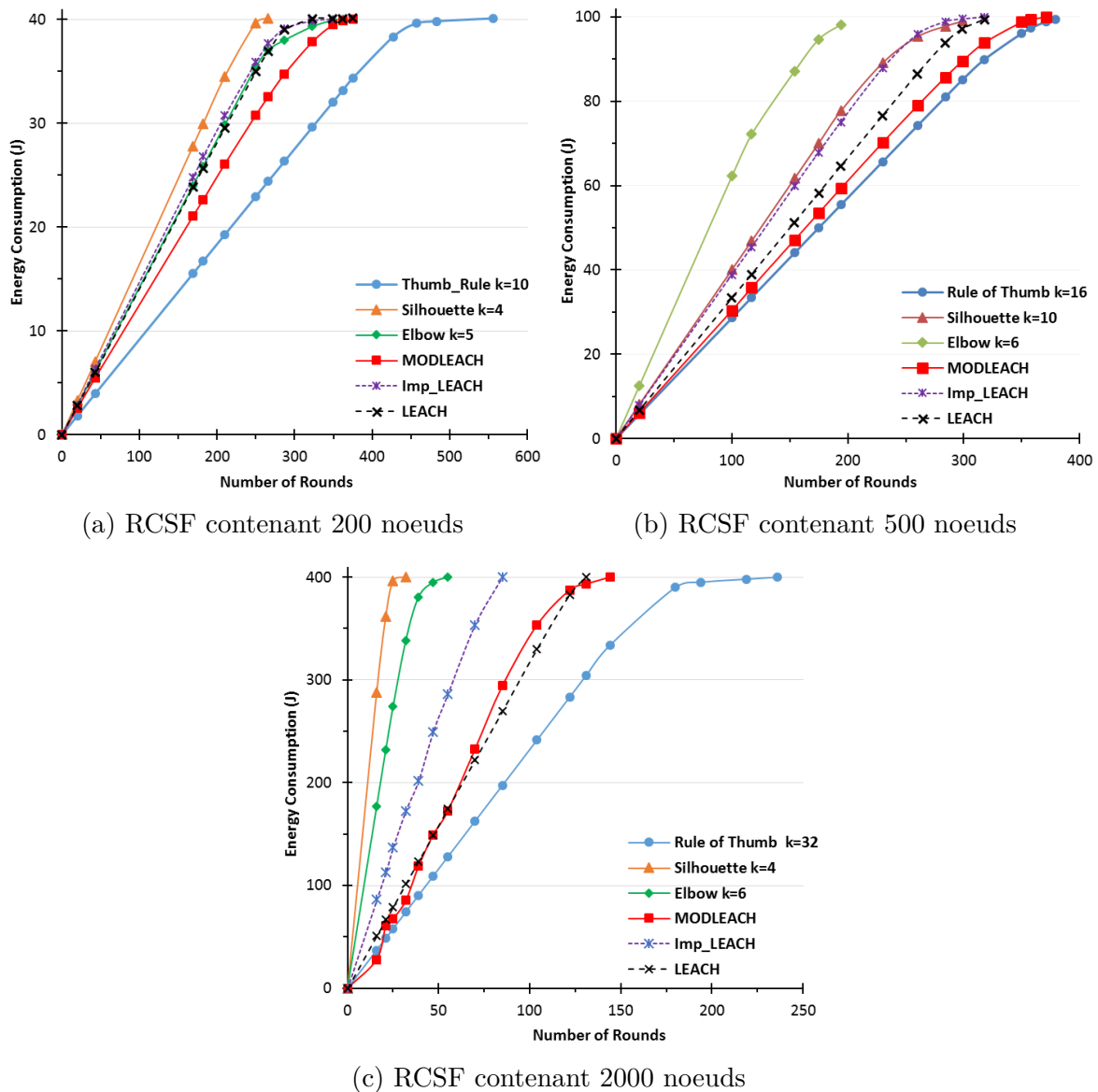


Figure 3.6: Consommation d'énergie en fonction de la taille des réseaux

a) Consommation d'énergie

La consommation d'énergie moyenne est illustrée par les Figures 3.6a, 3.6b et 3.6c. Ces figures montrent que lorsque le nombre de clusters varie en fonction de la taille du réseau, la consommation d'énergie diminue. Par conséquent, le schéma de routage basé sur la méthode de clustering "Rule of Thumb" présente une meilleure efficacité énergétique par rapport aux autres schémas de routage qui sont basés sur les autres méthodes de clustering. De plus, l'amélioration de la consommation d'énergie est satisfaite à la fois pour les CHs et pour les membres du cluster. Cette amélioration est obtenue grâce au bon choix du nombre de clusters par la méthode "Rule of Thumb" qui permet ainsi un bon équilibrage de charge en termes de consommation d'énergie dans le réseau. Les autres méthodes de clustering (Silhouette et Elbow) génèrent un nombre de clusters très réduit et ces clusters contiendront un grand nombre de noeuds membres et des CHs qui sont éloignés

de la station de base. De ce fait, les CHs épuisent rapidement leurs batteries en raison de leurs charges importantes (ils s'occupent d'un grand nombre de membres) d'une part et d'autre part puisqu'ils sont éloignés de la station de base ils dépensent plus d'énergie pour atteindre celle-ci lorsqu'ils lui transmettent les paquets de données agrégées. Par exemple, dans le réseau contenant 2 000 nœuds, l'énergie totale consommée par les nœuds est de (400 joules) après 256 rounds quand la méthode de clustering "Rule of Thumb" est utilisée, tandis que dans le meilleur cas pour les autres méthodes de clustering l'énergie totale du réseau est consommée après 144 rounds (c'est le cas du schéma de routage MODLEACH). En outre, pour les deux autres méthodes de clustering l'énergie de tous les nœuds était consommée après 32 rounds en raison de la charge importante supportée par les CHs pour servir leurs membres.

b) Durée de vie du réseau

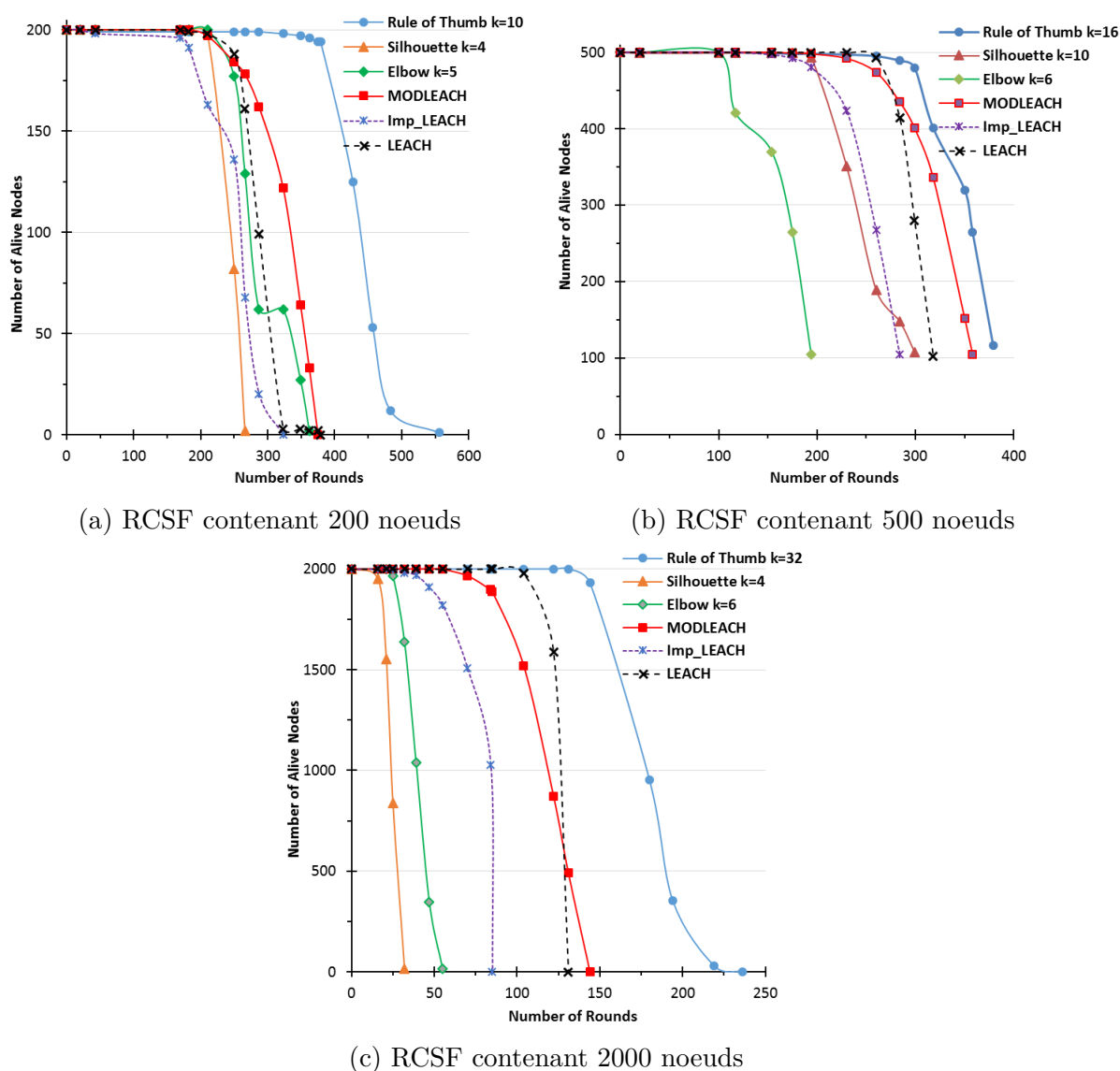


Figure 3.7: Durée de vie vs. Nombre de rounds

La durée de vie du réseau est définie comme le délai entre la mise en place du premier nœud jusqu'au décès du dernier nœud dans le premier scénario (réseau contenant 200 nœuds) et le délai écoulé entre la mise en place entre le premier nœud et la mort des 400 premiers nœuds dans le deuxième scénario (réseau contenant 500 nœuds).

Le critère de performance "Durée de vie du réseau" est très important dans les RCSFs car il est généralement impossible de remplacer et de recharger les batteries des nœuds lorsque ces derniers sont déployés dans des zones hostiles. Les Figures 3.7a, 3.7b et 3.7c représentent respectivement la durée de vie du réseau pour les différentes méthodes ("Rule of Thumb", Silhouette, Elbow, MODLEACH, imp_LEACH et LEACH) dans trois scénarios différents. Dans un réseau contenant 2000 nœuds (voir la Figure 3.7c), lorsque la méthode de clustering qui est basé sur "Rule of Thumb" est utilisée, le premier nœud meurt après 136 rounds alors que dans les autres méthodes: 13 rounds avec Silhouette, 21 rounds avec Elbow et Imp_LEACH, 49 rounds avec MODLEACH et 99 rounds avec LEACH. De plus, la méthode de clustering basée "Rule of Thumb" permet une plus longue durée de vie du réseau lorsque celui-ci est opérationnel pendant 236 rounds par rapport aux autres méthodes de clustering (32 rounds avec Silhouette, 55 rounds avec Elbow, 85 rounds avec Imp_LEACH, 131 rounds avec LEACH, 144 rounds avec MODLEACH). En outre, la méthode de clustering basée "Rule of Thumb" permet également un bon équilibrage de la consommation d'énergie tels que la période qui sépare la mort du premier nœud et la mort du dernier nœud est réduite dans les trois scénarios comme illustré par les Figures 3.7a, 3.7b et 3.7c.

c) Nombre de paquets envoyés à la station de base

Les Figures 3.8a, 3.8b et 3.8c illustrent le nombre de paquets envoyés à la station de base pendant la durée de vie du réseau. Le schéma de routage basé sur la méthode de clustering "Rule of Thumb" permet d'envoyer plus de paquets par rapport à ceux basés sur les méthodes Silhouette et Elbow pour les trois scénarios. Cependant, le schéma de routage basé sur l'approche de clustering de MODLEACH permet d'envoyer un grand nombre de paquets à la station de base dans les trois scénarios en raison du remplacement efficace des CHs et de la reconstruction du cluster.

La Figure 3.8a montre que le schéma de routage basé sur "Rule of Thumb" fournit de meilleurs résultats par rapport à LEACH et imp_LEACH alors que lorsque le nombre de nœuds augmente, comme illustré dans le deuxième et le troisième scénarios (Figures 3.8b et 3.8c), LEACH et imp_LEACH fournissent de meilleurs résultats que la méthode basée sur "Rule of Thumb" car lorsque la taille du réseau devient grande le nombre de clusters devient plus grand dans LEACH et imp_LEACH et par conséquent les CHs envoient plus de paquets à la station de base.

3.6 Conclusion

Dans ce chapitre, nous avons présenté une évaluation des performances des principales méthodes permettant de déterminer le meilleur nombre de clusters dans les réseaux de capteurs sans fil: "Rule of Thumb", Silhouette, Elbow et les méthodes de clustering utilisées dans LEACH, imp_LEACH et MODLEACH. Nous avons proposé cette évaluation pour trois scénarios avec des tailles de réseau différentes. Le premier et le deuxième scé-

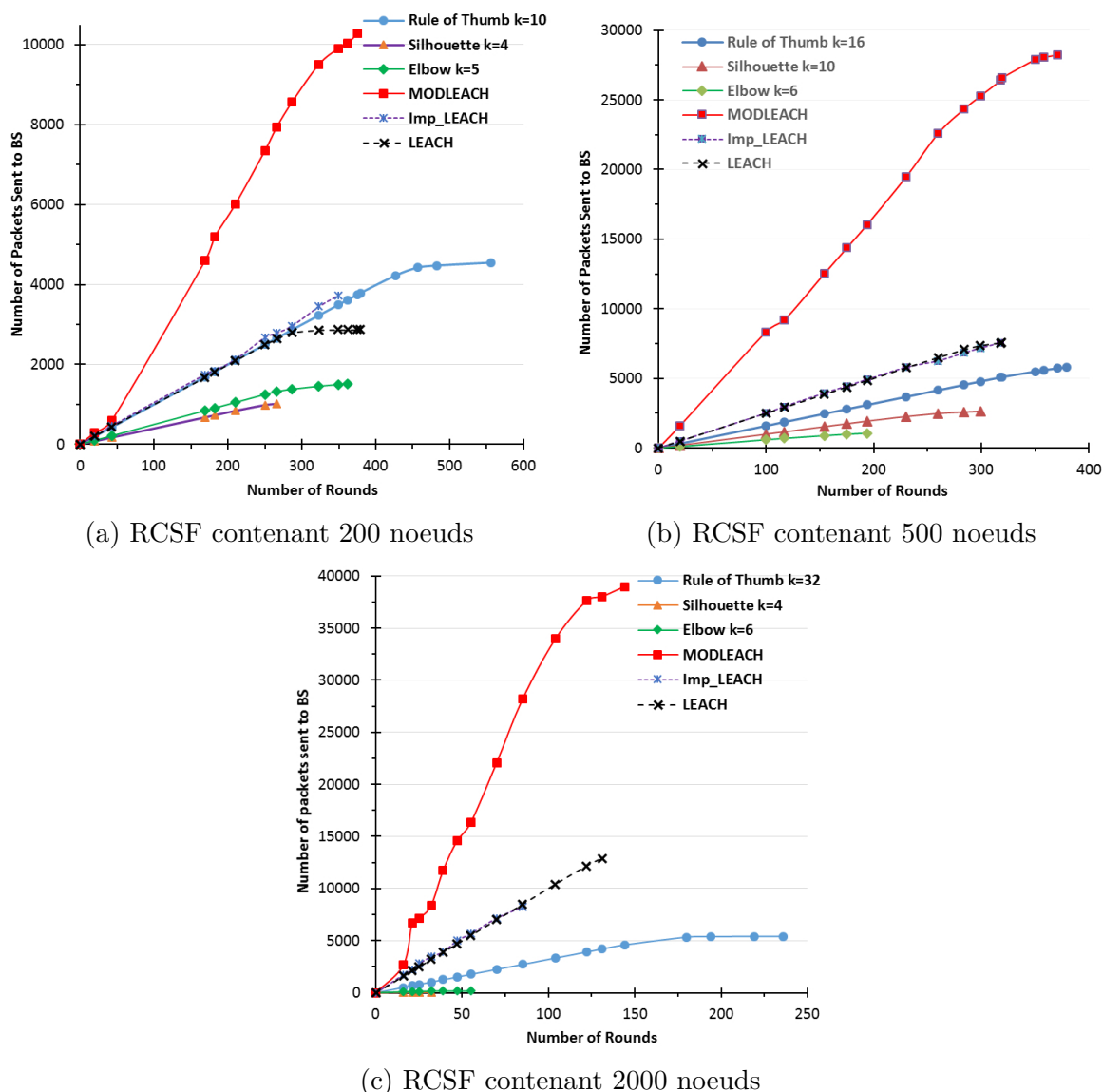


Figure 3.8: Nombre de paquets envoyés à la SB vs. Nombre de rounds

narios concernent un réseau de taille moyenne alors que le troisième est un réseau dense contenant un grand nombre de nœuds pour évaluer la robustesse de chaque méthode lors du passage à l'échelle.

Les résultats de simulations ont montré que le schéma de routage basé sur la méthode "Rule of Thumb" offre de meilleures performances en termes de durée de vie du réseau par rapport aux autres méthodes de clustering présentées dans ce chapitre, en particulier lorsque la taille du réseau est grande.

Nous proposons comme extension de ce travail un clustering dynamique pour chacune des méthodes présentées dans ce chapitre, suivie d'une évaluation des schémas de routage basés sur ces méthodes de clustering. De plus, nous proposons d'évaluer ces méthodes dans un environnement Internet des objets (IoT) où le nombre de nœuds peut être très grand.

Chapitre 4

Schémas de routage basés sur le
clustering pour les RCSFs à grande
échelle

Chapter 4

Schémas de routage basés sur le clustering pour les RCSFs à grande échelle

4.1 Introduction

Actuellement, les RCSFs sont devenus un outil incontournable dans la vie de tous les jours vu leur utilisation accrue dans divers domaines tels que le domaine militaire, la surveillance de l'habitat, l'observation des conditions environnementales, la santé, l'agriculture et la domotique.

Un RCSF est constitué généralement d'un grand nombre de nœuds de capteurs capables de détecter, de traiter, de stocker et de transmettre les données vers une ou plusieurs stations de base par le biais des communications sans fil [256]. Les capteurs sont généralement équipés de batteries à faible puissance et non rechargeables, et leur remplacement est difficile voire impossible surtout dans des environnements hostiles. De ce fait, l'efficacité énergétique est un problème majeur dans les RCSFs. Pour résoudre ce problème, les protocoles de routage basés sur le clustering sont considérés comme étant des protocoles prometteurs. Dans ce contexte, plusieurs protocoles ont été proposés dans la littérature [185, 257–263] montrant ainsi que le clustering a considérablement amélioré la durée de vie des RCSFs comparé aux solutions de routage à plat. De plus ce type de protocoles permettent le passage à l'échelle, l'agrégation/la fusion de données, moins de charge et plus de robustesse tout en optimisant la consommation d'énergie. Dans le processus de clustering, les capteurs sont regroupés en clusters et, pour chaque cluster, un nœud est choisi comme clusterhead (CH) selon certains critères. Ce dernier est responsable de la gestion de son cluster, de recevoir les paquets de données de ses membres, les agréger dans un seul paquet et le transmettre à la station de base [264].

Par ailleurs, nous pouvons classer les protocoles de routage basés sur le clustering en deux classes. La première classe s'intéresse au processus de clustering qui peut être soit statique soit dynamique. Dans le clustering dynamique, les clusters sont recréés selon des conditions bien précises durant la durée de vie des réseaux, par contre dans le clustering statique, les clusters sont formés une fois pour toute au long de la durée de la vie des réseaux [265–267]. La deuxième classe concerne l'entité responsable du clustering qui se compose de deux types : le clustering centralisé et le clustering décentralisé ou distribué.

Quand le processus de clustering est exécuté par la station de base il s'agit d'un clustering centralisé par contre lorsqu'il est exécuté par tous les noeuds du réseau il s'agit bien d'un clustering décentralisé ou distribué.

Dans les RCSFs basés sur le clustering, il y'a deux types de communications: les communications intra-cluster et les communications inter-cluster. Les communications intra-cluster auront lieu entre les capteurs d'un même cluster et les communications inter-cluster auront lieu entre deux membres de clusters différents du réseau. En outre, les communications peuvent être selon deux modes de transmission multi-sauts ou mono-saut.

Dans ce chapitre, nous proposons trois contributions économes en énergie pour l'extension de la durée de vie des RCSFs. Ces contributions font appel à la méthode K-Means pour structurer le réseau en clusters.

Dans la première contribution, nous proposons deux schémas de routage appelés "Dist-K-Means" et "Imp-Dist-K-Means" [27]. Le premier est basé sur la version standard de K-Means alors que le deuxième schéma est basé sur une version améliorée de K-Means. Le deuxième schéma de routage génère des clusters équilibrés ce qui permet de distribuer la charge entre CHs équitablement, contrairement à LEACH [159], où l'un des clusters générés peut contenir un grand nombre de noeuds et un autre avec un petit nombre de noeuds. De plus, dans cette contribution l'élection des CHs se fait d'une manière distribuée après chaque round en fonction de l'énergie résiduelle des noeuds et de leurs distances vers les centroïdes.

Dans la deuxième contribution, nous visons à améliorer le processus de clustering en proposant une solution appelée "Cond-Dist-K-Means" [28]. Cette solution implique un clustering dynamique dans lequel le processus de clustering est déclenché à chaque fois que le pourcentage de noeuds morts dépasse 5% des noeuds vivants. A cet effet, le nombre de clusters est recalculé avec la méthode Silhouette [22] et de nouveaux clusters seront créés en utilisant la méthode K-Means.

Dans la troisième contribution, nous proposons un nouveau protocole de routage appelé "GA-Dist-K-Means" pour prolonger la durée de vie dans les RCSFs [268]. Ce schéma de routage est basé sur une version améliorée des algorithmes génétiques et sur la méthode K-Means. La méthode K-Means est utilisée pour créer des clusters et les clusterheads sont élus en fonction de la distance, de l'énergie résiduelle et du nombre de fois qu'un noeud était élu comme CH tandis que l'algorithme génétique est appliqué pour établir les communications inter-cluster. Dans GA-Dist-K-Means, le processus de clustering est relancé si tous les membres d'un même cluster sont morts et le processus de construction de chemins est déclenché quand un noeud change d'état, c'est-à-dire qu'un autre noeud membre devient CH.

Ces trois contributions ont été développées sous MATLAB, évaluées et comparées à d'autres protocoles. Les résultats obtenus ont montré que nos contributions apportaient des améliorations significatives en termes de consommation d'énergie et de durée de vie des réseaux par rapport à d'autres protocoles.

4.2 Contexte

Les critères de performances d'un protocole de routage conçu pour un RCSF dépendent des besoins de son application. Il existe des applications qui nécessitent le déploiement

d'un grand nombre de capteurs avec une forte densité dans la zone d'intérêt. Néanmoins, l'efficacité énergétique reste l'un des principaux défis dans les RCSFs en raison de la contrainte énergétique des capteurs [269]. Une des techniques qui permet à la fois l'efficacité énergétique et le passage à l'échelle est le clustering. Dans ce contexte, nous avons utilisé deux étapes principales lors de la conception des protocoles de routage qui sont l'organisation du réseau et l'acheminement des données. Durant l'organisation du réseau, le processus de clustering est utilisé pour permettre une efficacité énergétique et minimiser la charge du réseau lors du passage à l'échelle.

Dans ce qui suit, nous décrivons les travaux connexes à cette problématique, nous détaillons chaque contribution et nous présentons les résultats obtenus.

4.3 Travaux connexes

Les recherches récentes dans les RCSFs ont conduit à la conception de plusieurs protocoles de routage hiérarchiques ou basés sur le clustering permettant de minimiser la consommation d'énergie et par conséquent prolonger la durée de vie des réseaux [150]. La consommation d'énergie des capteurs est principalement dominée par les tâches de communications c'est-à-dire les transmissions et les réceptions. A cet effet, les protocoles de routage dans les RCSFs doivent se focaliser sur un routage économe en énergie pour prolonger la durée de vie des capteurs et donc des réseaux. Dans cette section, nous nous sommes focalisés sur les protocoles de routage basés sur le clustering.

LEACH [24] est un protocole de routage hiérarchique conçu pour les RCSFs. Il est considéré comme un point de repère pour n'importe quelle nouvelle contribution faisant appel au clustering. Il se compose de plusieurs rounds et chaque round se compose en deux phases: la phase de formation des clusters (setup) et la phase de communication des données (steady-state). Il utilise un clustering distribué où chaque nœud tente d'être CH selon un modèle de probabilité en générant un nombre aléatoire compris entre 0 et 1. Si ce nombre est inférieur au seuil $T(n)$ ce nœud devient CH. Ensuite, ce CH envoie un message ACK aux nœuds non-CH pour leur aviser de son nouveau statut. Les nœuds non-CHs envoient un message pour demander de rejoindre un CH en se basant sur la force du signal reçu et par conséquent les clusters seront créés. Dans la deuxième phase, chaque nœud membre d'un cluster transmet les données collectées pendant son intervalle de temps (time-slot) à son CH correspondant, puis le CH agrège toutes les données reçues en un seul paquet de données et le transmet directement à la station de base.

Dans [229], les auteurs ont présenté un protocole basé sur un clustering distribué impliquant la méthode K-Means. Dans ce protocole, le processus de clustering est effectué par chaque nœud du réseau. Les résultats ont montré que ce protocole est plus efficace que le clustering centralisé en termes de rapidité du processus de clustering mais la consommation d'énergie est presque la même dans les deux approches (distribuée et centralisée).

KEAC [231] est un protocole de routage centralisé basé sur le clustering pour les RCSFs. Dans ce protocole, la méthode K-means est utilisée pour la formation des clusters et l'élection des CHs est fait en fonction de l'énergie résiduelle des nœuds et de la distance qui les sépare de leurs centroïdes. Le protocole KEAC consomme moins d'énergie et prolonge plus la durée de vie des réseaux par rapport à LEACH [24] et LEACH-C [159].

Dans [233], les auteurs ont proposé deux approches de clustering basées sur K-Means.

La première implique la distance dans le processus d'élection des CHs alors que la deuxième implique la distance et l'énergie restante des noeuds dans ce processus d'élection des CHs. Les résultats de simulations ont montré que la deuxième approche est la meilleure en termes de consommation d'énergie et de durée de vie des réseaux comparée à la première et à LEACH [24] alors que la première approche surpasse LEACH.

Dans [234], les auteurs ont proposé une version améliorée de LEACH nommée KM-LEACH dans laquelle le processus de clustering est centralisé. KM-LEACH implique la méthode K-Means pour la sélection des CHs en favorisant les noeuds les proches des centroïdes et qui ont une énergie résiduelle supérieure à un seuil spécifié d'être des CHs. Après cette phase de sélection des CHs, il y a l'occurrence de la phase de formation des clusters et la phase de communication des données à la station de base qui se font de la même manière que dans LEACH. Les résultats de simulations ont illustré que KM-LEACH réduit la consommation d'énergie et prolonge la durée de vie des RCSFs par rapport à LEACH.

Dans [239], les auteurs ont proposé un protocole hybride basé sur K-Means et sur LEACH pour les RCSFs. Ce protocole implique la méthode K-Means pour former les clusters et le protocole LEACH dans chaque cluster pour la sélection des CHs. Les résultats de simulations ont montré que ce protocole surpasse LEACH en termes de consommation d'énergie et de durée de vie, mais l'inconvénient de ce protocole réside dans la consommation d'énergie qui est élevée pendant le processus d'élection des CHs.

Dans [270], les auteurs ont proposé un protocole de routage appelé KMV-LEACH qui implique la méthode K-medoids dans le processus de clustering. Dans KMV-LEACH, le protocole LEACH est utilisé pour la sélection des CHs. Par ailleurs, les auteurs ont fait une comparaison entre Gap_statistique, CalinskiHarabasz et la méthode Silhouette afin de déterminer le nombre optimal de clusters. Les résultats ont montré que la méthode Silhouette minimise la consommation d'énergie et prolonge la durée de vie des réseaux par rapport aux deux autres méthodes.

Dans [230], un protocole de routage basé sur le clustering appelé LEACH-KED a été proposé. Dans ce protocole, le processus de clustering est effectué après chaque round par la méthode K-Means et la sélection des CHs est réalisée en fonction de l'énergie, de la distance par rapport au centroïde et de la distance par rapport à la station de base. Les résultats de simulations ont montré que ce protocole surpasse LEACH [24] et LEACH-C [159] en termes de consommation d'énergie.

Dans [271], les auteurs ont proposé une approche visant à trouver le nombre de clusters optimal dans les RCSFs avec une station de base mobile. Dans cette approche, la station de base se déplace vers les CHs pour collecter des données agrégées. Cette approche permet une atténuation nulle lors de la collecte de données à partir des CHs par la station de base. Cependant, élire les CHs durant toute la vie du réseau peut avoir un impact négatif sur sa durée de vie puisque les CHs sont responsables de plusieurs tâches pouvant épuiser leurs batteries rapidement et donc rendre le réseau inutilisable.

Dans [149], les auteurs ont proposé une nouvelle combinaison de la méthode K-Means avec une amélioration des algorithmes génétiques (AGs) pour réduire la consommation d'énergie et prolonger la durée de vie des réseaux. Ce schéma de routage réduit la consommation d'énergie en trouvant le nombre optimal des CHs à l'aide des AGs et permet une longue durée de vie des réseaux par rapport aux protocoles LEACH [24], GAEEP [272] et GABEEC [138].

ROS_IGA [273] vise à prolonger la durée de vie des réseaux avec une consommation d'énergie optimale. Ce protocole utilise l'énergie résiduelle des nœuds, la distance entre les nœuds adjacents, le délai de communication et les nœuds relais comme paramètres de l'AG. En outre, la nouveauté de ROS_IGA est d'assurer la conformité entre la topologie du RCSF et la demande de communication entre les nœuds. Cela est réalisé par la résolution du problème de chromosomes invalides en utilisant un algorithme génétique amélioré. Les résultats de simulations ont montré que ROS_IGA offre un meilleur routage que celui du protocole d'inondation proposé dans [274].

Dans le protocole "GAR" [142], les auteurs ont traité le problème de routage des données de tous les nœuds relais vers la station de base via d'autres nœuds relais en proposant un protocole de routage basé sur l'AG. Ce protocole prend en compte le problème de consommation d'énergie en minimisant la distance totale parcourue à chaque round. GAR a amélioré la durée de vie des réseaux par rapport à MHRM [143]; mais il n'a pas pris en considération l'énergie résiduelle des nœuds sinon les résultats seront beaucoup meilleurs.

Les auteurs de [275] ont visé à minimiser la consommation d'énergie dans les RCSFs en impliquant les AGs dans le processus de routage. L'utilisation des AGs a pour but d'optimiser et de trouver les chemins entre les nœuds vers ou depuis la station de base tout en minimisant l'énergie consommée. Le protocole proposé prolonge la durée de vie des réseaux par rapport au protocole TEEN [8].

Les protocoles présentés dans cette section, n'impliquent pas conjointement une approche centralisée pour former les clusters, une approche distribuée pour l'élection des CHs et une approche pour établir des chemins entre les CHs. Dans nos contributions, nous proposons d'organiser le réseau en clusters et d'acheminer les données à la station de base en impliquant plusieurs approches telles la méthode K-Means, la méthode Silhouette, la méthode "Rule of Thumb" et les algorithmes génétiques pour prolonger la durée de vie des réseaux et pour permettre le passage à l'échelle.

4.4 Travaux proposés

La conservation de l'énergie est l'objectif principal dans les RCSFs. Ainsi, pour atteindre cet objectif et prolonger la durée de vie des réseaux, nous proposons trois contributions dans lesquelles nous faisons appel au concept de clustering. Ces contributions sont des schémas de routage économes en énergie pour les RCSFs où chacun d'eux implique une méthode intelligente ou une métaheuristique pour déterminer le nombre de clusters approprié et/ou la manière d'élire les CHs ainsi que la manière d'établir les chemins entre les nœuds et la station de base.

Par ailleurs, plusieurs études ont abordé le problème de la consommation d'énergie, et aussi d'autres recherches ont traité le problème du passage à l'échelle dans les réseaux de capteurs c'est pourquoi les chercheurs ont proposé l'architecture hiérarchique pour résoudre ce type de problèmes [25].

Dans ce qui suit, tout d'abord nous présentons les hypothèses sur lesquelles se basent nos contributions, ainsi que l'outil de simulation et le modèle d'énergie. Puis, nous décrivons en détails nos contributions et nous évaluons les performances des schémas de routage proposés sous-jacents à chacune des contribution et nous les comparons avec LEACH [24] en termes de consommation d'énergie et de durée de vie des réseaux pour illustrer leur robustesse.

4.4.1 Hypothèses

Dans ce chapitre pour les trois contributions, nous supposons que:

- Les nœuds sont déployés d'une manière aléatoire dans l'environnement et ils sont statiques.
- Les nœuds peuvent ajuster leur puissance de transmission en fonction de la distance qui les sépare du nœud destinataire.
- Au départ, tous les capteurs ont le même niveau énergétique.
- Les identifiants (ID) et les positions de tous les nœuds sont connus par la station de base.
- Chaque nœud connaît sa propre position.
- La station de base n'a pas de contrainte énergétique.

4.4.2 Outil de simulation

Les résultats de simulation de nos contributions sont obtenus à l'aide de MATLAB qui est un environnement de simulation permettant de combiner la programmation textuelle et graphique pour concevoir nos scénarios. En outre, l'environnement MATLAB nous permet d'utiliser directement des milliers d'algorithmes déjà présents. Les paramètres utilisés lors de la simulation sont différentes pour les trois contributions. Nous avons répété 10 fois la simulation pour chaque scénario associé à une telle contribution et la moyenne de ces répétitions est présentée dans les résultats.

4.4.3 Modèle d'énergie radio

Plusieurs recherches ont été effectuées dans le développement des modèles à faible dissipation d'énergie. Dans nos travaux, nous avons utilisé le même modèle que celui décrit dans [24] où une partie de l'énergie est dissipée par le récepteur pour faire fonctionner le composant électronique associé à la radio et celle de l'émetteur pour faire fonctionner le composant électronique associé à la radio et l'amplificateur de puissance.

A cet effet, la consommation d'énergie est principalement divisée en deux parties: les réceptions et les transmissions. La consommation d'énergie des transmissions nécessite une énergie supplémentaire pour amplifier le signal en fonction de la distance qui le sépare de la destination. Ainsi, pour transmettre un message de k bits à une distance d , la radio dépense de l'énergie comme décrit par l'équation 4.1, où ε_{elec} est l'énergie consommée par le composant électronique associé à la radio, ε_{friss_amp} et $\varepsilon_{two_ray_amp}$ pour amplifier le signal. La consommation d'énergie des réceptions est exprimée par l'équation 4.2.

$$E_{Tx}(k, d) = \begin{cases} k * E_{elec} + \varepsilon_{fs} * d^2, & \text{if } d < d_0 \\ k * E_{elec} + \varepsilon_{amp} * d^4, & \text{if } d \geq d_0 \end{cases} \quad (4.1)$$

$$E_{Rx}(k) = k * E_{elec} \quad (4.2)$$

4.4.4 Contribution 1: Schémas de routage Dist-K-Means et Imp-Dist-K-Means

Dans cette contribution, nous avons proposé deux schémas de routage basés sur K-Means (Dist-K-Means et Imp-Dist-K-Means) dont le processus d'élection des CHs est réalisé d'une manière distribuée [27]. Le premier schéma (Dist-K-Means) génère des clusters avec la version standard de K-Means, tandis que le second (Imp-Dist-K-Means) génère des clusters avec une version améliorée de K-Means.

Le schéma de routage *Dist-K-Means* est établi selon l'algorithme 1 alors que le schéma de routage *Imp-Dist-K-Means* est établi selon l'algorithme 2.

Algorithm 1 Dist-K-Means

Require: (Network size, Node positions)

Ensure: Execution of Dist-K-Means

```
1: if (round = 1) then
2:   Determination of clusters number
3:   Find of initial centroids's positions based on the density of the nodes
4:   Formation of clusters
5:   The sink sends to every node in the network its affiliation
6: end if
7: while (the network is functional) do
8:   Election of CHs in a distributed manner based on Energy and Distance to centroid

9:   Data transfer
10: end while
```

Algorithm 2 Imp-Dist-K-Means

Require: (Network size, Node positions)

Ensure: Execution of Imp-Dist-K-Means

```
1: if (round = 1) then
2:   Determination of clusters number
3:   Formation of clusters with Imp-K-Means
4:   The sink sends to every node in the network its affiliation
5: end if
6: while (the network is functional) do
7:   Election of CHs in a distributed manner based on Energy and Distance to centroid

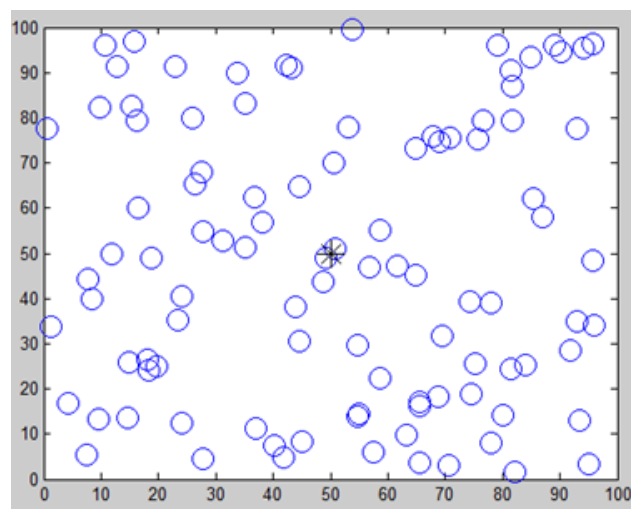
8:   Data transfer
9: end while
```

a) Présentation de Dist-K-Means et Imp-Dist-K-Means

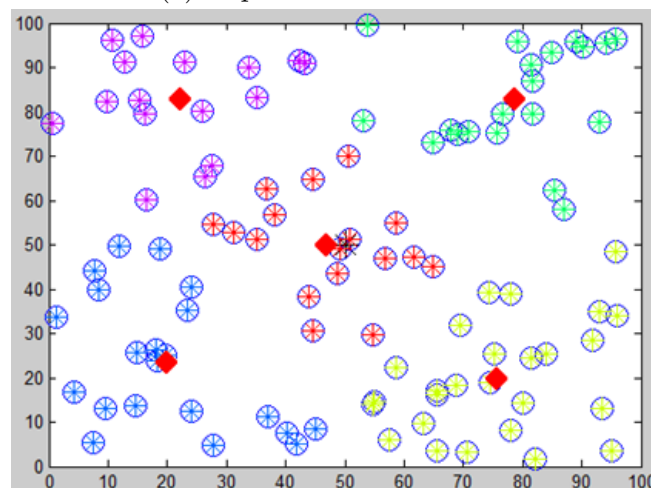
Ces deux schémas de routage se composent de deux phases: la phase de configuration (set-up) et la phase d'exécution. La deuxième phase est la même dans les deux schémas de routage proposés.

Phase setup du schéma de routage Dist-K-Means

Dans ce schéma de routage basé sur le clustering, la station de base commence par collecter les positions de tous les nœuds du réseau à l'aide d'un système de localisation GPS, ensuite elle exécute la méthode K-Means après avoir déterminé le nombre de clusters et les positions des centroïdes initiaux. Lorsque le processus de clustering est terminé, la station de base envoie un message de contrôle à chaque nœud du réseau pour lui informer à quel cluster il est affilié. La Figure 4.1a illustre le déploiement aléatoire des nœuds dans une zone d'intérêt et la Figure 4.1b montre la formation des clusters après l'exécution de cette phase où les carrés rouges représentent les centroïdes. Nous constatons que les clusters générés ne sont pas équilibrés en termes de nombre de nœuds, ce qui peut entraîner une charge supplémentaire pour les CHs dans le cas où leurs clusters contiennent un grand nombre de membres.



(a) Déploiement des noeuds



(b) Clusters formés

Figure 4.1: Formation des clusters par K-Means

Phase setup du schéma de routage Imp-Dist-K-Means

Le mécanisme de clustering utilisé par ce schéma de routage comprend deux sous-phases:

la sous-phase de préparation et la sous-phase de ré-affiliation. La première sous-phase est similaire à la phase setup du premier schéma de routage décrit précédemment et la seconde sous-phase consiste en un processus de ré-affiliation des nœuds pour générer des clusters équilibrés, c'est-à-dire des clusters de même taille. Ainsi, après le processus de formation des clusters et le calcul du nombre moyen de membres dans les clusters, les membres appartenant à des clusters dont la taille est supérieure à la moyenne et se trouvant aux frontières de ces clusters rejoignent des clusters dont la taille n'a pas atteint la taille moyenne. De plus, ces membres choisissent les clusters en fonction de la distance qui les sépare de leurs centroïdes. La Figure 4.2 (les carrés rouges représentent les centroïdes) illustre la formation des clusters par cette approche. Ce processus de clustering génère des clusters équilibrés, ce qui permet l'équilibrage de la charge entre les CHs et par conséquent une meilleure gestion de la consommation d'énergie .

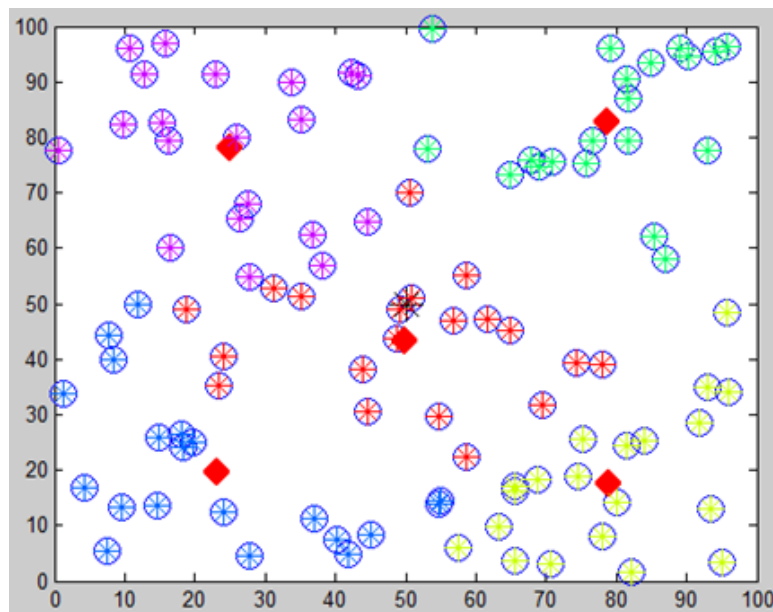


Figure 4.2: Clusters générés par l'approche Imp-Dist-K-Means

Phase d'exécution

Cette phase est réalisée en plusieurs rounds qui représentent la durée de la phase d'exécution. Chaque round comprend à son tour deux sous-phases: la sous-phase de l'élection des CHs et la sous-phase de transmission de données. Au cours de la première sous-phase, dans chaque cluster, un CH est élu comme suit: chaque membre du cluster calcule son poids et l'envoie à tous les membres du cluster auquel il appartient. Le poids d'un nœud est calculé en fonction de son énergie résiduelle et la distance qui le sépare du centroïde de son cluster correspond comme montre l'équation 4.3. Le nœud qui a le plus grand poids, devient CH pendant ce round. Ensuite, chaque CH envoie un time-slot selon l'approche TDMA à chaque membre de son cluster. Après cela, la deuxième sous-phase commence et au cours de laquelle chaque membre du cluster envoie les données collectées à son CH correspondant selon son time-slot. Après réception de tous les paquets par le CH, ce dernier les regroupe en un seul paquet et le transmet à la station de base.

$$Cost(Node_i) = \alpha * Energy + \frac{\beta}{dist(Node_i, Centroid)} \quad (4.3)$$

où

$$\alpha + \beta = 1 \quad \alpha \in]0.75, 0.95[\quad \beta \in]0.05, 0.25[$$

b) Évaluation et résultats de simulations

Dans cette section, nous allons évaluer les performances des schémas de routage que nous avons proposé en les comparant avec LEACH [159] en termes de consommation d'énergie et de durée de vie des réseaux.

Environnement de simulation

Nous avons réalisé les simulations à l'aide du simulateur Matlab. Les paramètres utilisés lors des simulations sont résumés dans le tableau 4.1. Nous avons répété 10 fois les simulations et la moyenne de ces répétitions est présentée dans les résultats.

Table 4.1: Paramètres de simulation (Dist-K-Means et Imp-Dist-K-Means)

Paramètres	Valeurs
Grille du réseau	(0,0)x(100,100)
Position de la station de base	(50,50)
ε_{elec}	50 nJ/bit
ε_{friss_amp}	10pJ/bit/m ²
$\varepsilon_{two_ray_amp}$	0.0013pJ/bit/m ⁴
d_0	87 m
Énergie initiale par noeud	0.2 J
Nombre de noeuds	100, 200
P	0.05

Résultats de simulations

Pendant une période, chaque membre n'envoie qu'un seul paquet de données à son CH correspond et à la fin de cette période, chaque CH agrège tous les paquets de données reçus de ses membres en un seul paquet et l'envoie à la station de base. Par ailleurs, le poids d'un nœud est calculé en fonction de son énergie résiduelle et de la distance qui le sépare du centroïde le plus proche, comme indiqué dans l'équation 4.3. Par ailleurs, les coefficients α et β associés à ces deux paramètres (énergie résiduelle et distance) doivent être bien choisis pour permettre de meilleures performances. Pour cela, nous avons varié ces deux coefficients pour trouver les meilleures valeurs qui fournissent de meilleures performances. La Figure 4.3 montre que lorsque nous favorisons la métrique énergie sur la métrique distance, les résultats sont meilleurs.

Pour évaluer les performances des deux schémas de routage proposés dans cette première contribution, nous avons affecté à α la valeur 0,95 et à β la valeur 0,05. Les Figures 4.4 et 4.5 montrent que les techniques de clustering proposées basées sur l'approche K-Means (Dist-K-Means et Imp-Dist-K-Means) offrent de meilleures performances que celles

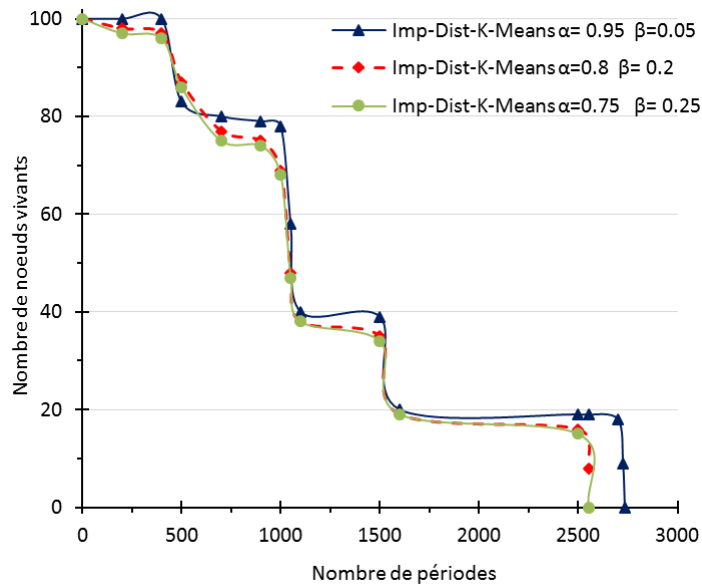


Figure 4.3: Variation de α and β dans Imp-Dist-K-Means

de LEACH en raison de l'approche distribuée utilisée lors de l'élection des CHs. De plus, la deuxième approche proposée (Imp-Dist-K-Means) offre de bonnes performances car la charge des CHs est bien équilibrée puisque les clusters générés ont presque la même taille.

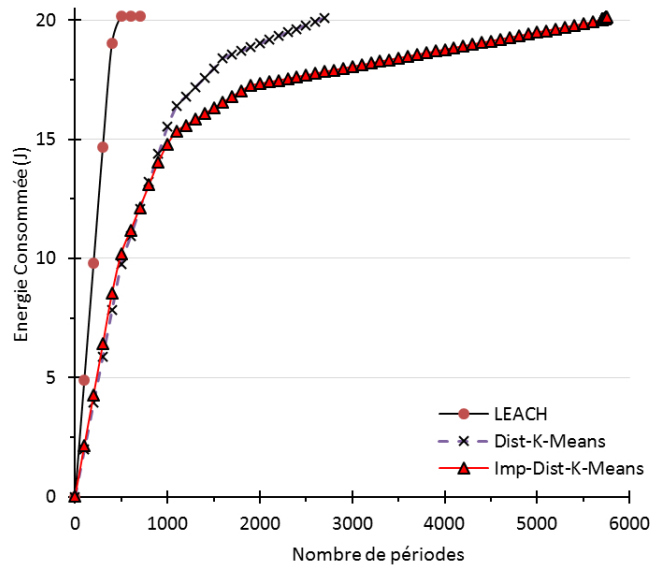


Figure 4.4: Consommation d'énergie vs. Nombre de périodes (100 noeuds)

Pour illustrer la robustesse des schémas de routage proposés lorsque le nombre de nœuds augmente dans le réseau, nous avons évalué les performances de ces trois schémas de routage pour des réseaux contenant 200 nœuds et nous avons comparé les résultats obtenus. De plus, comme la simulation prend beaucoup de temps, nous avons considéré que la durée de vie du réseau est définie par la mort de la moitié des nœuds (50% des nœuds). Les Figures 4.6 et 4.7 montrent également que les performances de Imp-Dist-K-

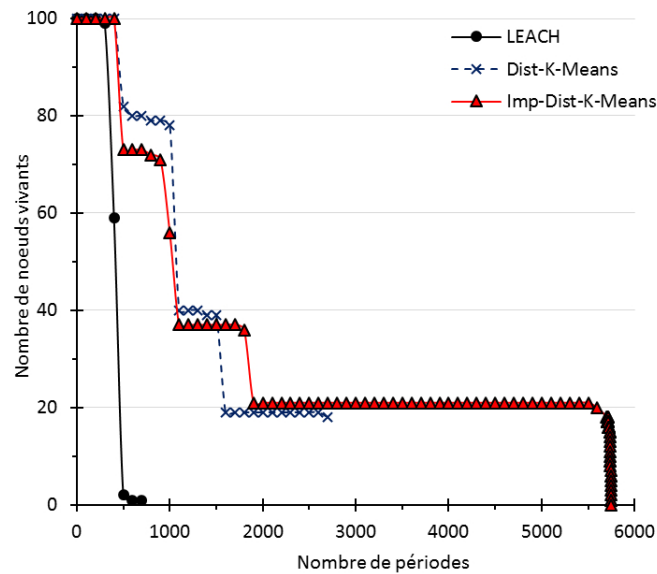


Figure 4.5: Nombre de noeuds vivants vs. Nombre de périodes (100 noeuds)

Means sont toujours supérieures à celles des schémas de routage LEACH et Dist-K-Means.

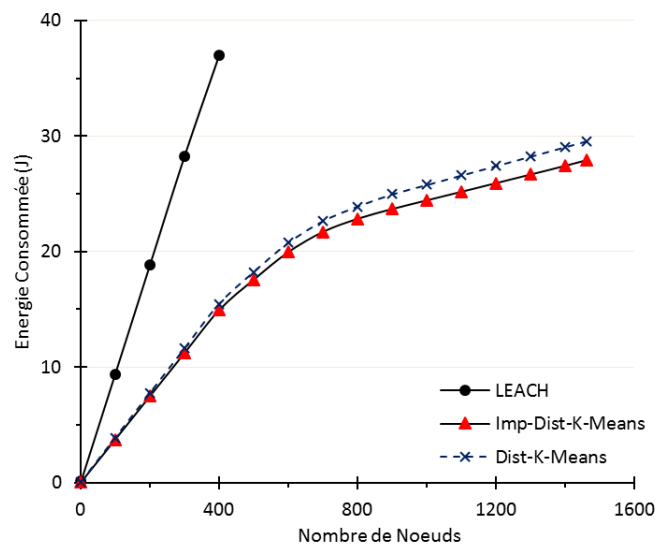


Figure 4.6: Consommation d'énergie vs. Nombre de périodes (200 Noeuds)

4.4.5 Contribution 2: Schéma de routage Cond-Dist-K-Means

Le problème principal dans les RCSFs est l'optimisation de la consommation de l'énergie. L'approche de clustering est une solution prometteuse et tellement utilisée, qui peut prolonger la durée de vie des réseaux. Dans cette contribution appelée "Cond-Dist-K-Means" [28], nous visons à améliorer le processus de clustering. La solution proposée utilise la méthode Silhouette pour déterminer le nombre optimal de clusters et K-Means pour la formation des clusters et dans laquelle le processus de clustering est effectué sous des conditions prédéterminés. Ce processus de clustering sera déclenché à chaque fois que le

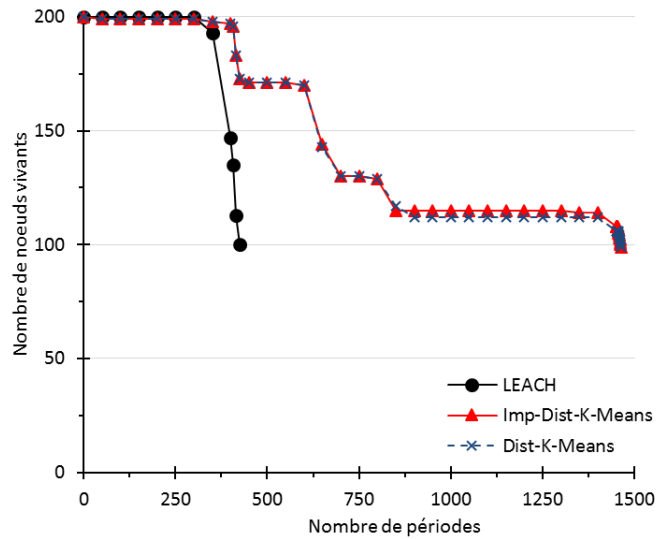


Figure 4.7: Nombre de noeuds vivants vs. Nombre de périodes (200 Noeuds)

pourcentage de noeuds morts dépasse 5% des noeuds vivants. A cet effet, le nombre de clusters est recalculé en utilisant la méthode Silhouette [22] et les clusters sont créés en utilisant l'approche K-Means [23].

Dans ce qui suit, nous détaillons la contribution proposée *Cond-Dist-K-Means* et nous évaluons ses performances dans différents scénarios en particulier dans des scénarios là où le nombre de noeuds dans un réseau est assez grand pour illustrer sa robustesse lors du passage à l'échelle. Puis, nous comparons les résultats obtenus avec d'autres protocoles de routage tels que le protocole de routage Dist-K-Means et le protocole LEACH.

a) Présentation de Cond-Dist-K-Means

Dans cette section, nous décrivons l'essentiel de notre contribution qui consiste en un protocole de routage distribué et dynamique basé sur la méthode Silhouette [22] pour déterminer le nombre approprié de clusters et la méthode K-Means [23] pour former les clusters. Le schéma de routage proposé comprend deux phases: la phase de configuration (setup) et la phase de communication des données.

Le schéma de routage *Cond-Dist-K-Means* est établi selon l'algorithme 3.

Au cours de la phase setup, la station de base collecte les informations et les positions de tous les noeuds du réseau à l'aide d'un système de localisation tels que le GPS et effectue le clustering grâce à la méthode K-Means après la détermination du nombre de clusters avec la méthode Silhouette et les positions initiaux des centroïdes. Lorsque le processus de clustering est achevé, la station de base envoie un message à chaque capteur pour l'informer à quel cluster il est affilié. La Figure 4.8a illustre le déploiement aléatoire des noeuds dans une zone d'intérêt et la Figure 4.8b montre la formation de quatre clusters après l'exécution de la méthode Silhouette et de la méthode de clustering K-Means.

Le schéma de routage proposé est réalisé en plusieurs périodes (rounds) qui représentent la durée de la phase setup et de la phase de communication de données. La phase setup comprend trois sous-phases: la sous-phase de la détermination du nombre de clusters par la méthode Silhouette, la sous-phase de re-clustering exécutée par la méthode

Algorithm 3 Cond-Dist-K-Means

Require: (Network size, Node positions, Threshold)**Ensure:** Execution of Cond-Dist-K-Means

```

1: if (round = 1) then
2:   Determination of clusters number
3:   Formation of clusters with K-Means
4:   The sink sends to every node in the network its affiliation
5: end if
6: while (the network is functional) do
7:   if (percentage of mortality exceeds the Threshold) then
8:     Determination of clusters number
9:     Formation of clusters with K-Means
10:    The sink sends to every node in the network its affiliation
11:   end if
12:   Election of CHs in a distributed manner based on Energy and Distance to centroid

13:   Data transfer
14: end while

```

K-Means et la sous-phase d'élection des CHs. Au cours de la sous-phase d'élection des CHs, dans chaque cluster, un CH est élu d'une manière distribuée comme dans la première contribution. Ensuite, chaque CH envoie un échéancier selon l'approche TDMA (des time-slots) à tous les membres de son cluster correspondant. La phase setup est déclenchée lorsque le pourcentage de nœuds vivants diminue d'un certain seuil fixe (5%) et pendant laquelle les méthodes K-Means et Silhouette sont exécutées pour restructurer de nouveau le réseau en clusters.

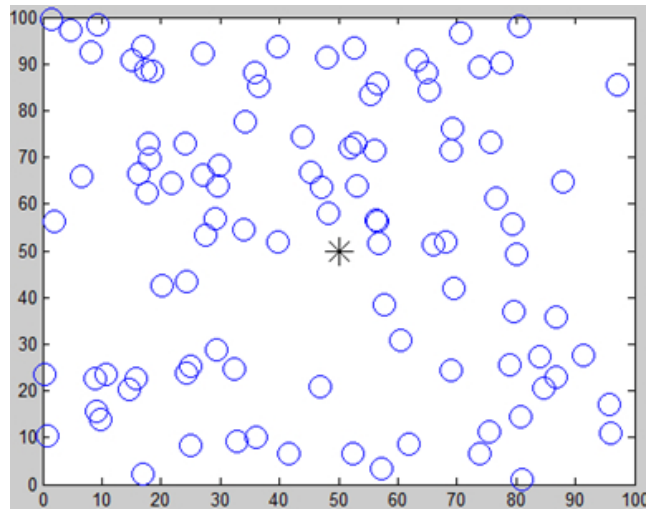
Lorsque le processus de clustering est achevé, la station de base envoie un message à chaque nœud pour l'informer de sa nouvelle affiliation. La Figure 4.9 montre la formation de six clusters après l'exécution de la phase setup. Enfin, la phase de communication de données commence et pendant laquelle chaque nœud membre envoie les données collectées à son CH correspondant selon son time-slot TDMA en un seul saut. Après avoir reçu tous les paquets par le CH, ce dernier les regroupe en un seul paquet et le transmet directement à la station de base.

b) Evaluation des performances de Cond-Dist-K-Means

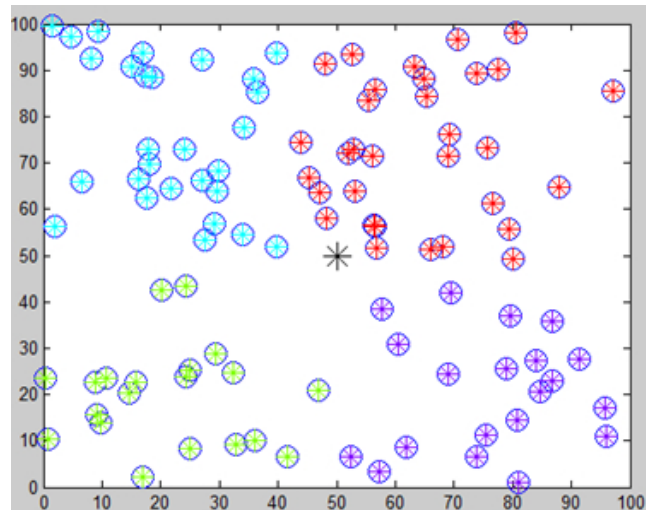
Dans cette section, nous évaluons le schéma de routage proposé dans cette contribution et nous comparons ses performances au schéma de routage Dist-K-Means [27] et au protocole LEACH [159] en termes de consommation d'énergie et de durée de vie des réseaux.

Environnement de simulations

Toutes les simulations ont été effectuées à l'aide du simulateur MATLAB. Nous avons utilisé deux scénarios pour évaluer les performances de notre contribution. Le premier scénario consiste en un réseau de 100 nœuds déployés d'une manière aléatoire dans une zone de 100 m x 100 m et la station de base est placée au point (50,50). Le deuxième scénario se compose de 500 nœuds répartis d'une manière aléatoire dans une zone de



(a) Déploiement des nœuds



(b) Clusters formés

Figure 4.8: Formation des clusters par Cond-Dist-K-means

200 m x 200m et la station de base est située au centre de la zone d'intérêt c'est-à-dire au point (50,50). Le second scénario est utilisé pour illustrer la robustesse de notre contribution lorsque le nombre de nœuds augmente dans le réseau. Dans ces simulations, nous avons supposé que chaque nœud transmet un paquet de données par round à son CH correspondant et le nombre de clusters dans LEACH représente 5% du nombre total de nœuds du réseau ($P = 5\%$). Les paramètres utilisés dans les simulations sont résumés dans le tableau 4.2.

Résultats de simulations

Dans ce qui suit, nous présentons les résultats obtenus lors de l'évaluation des performances de notre contribution et nous les comparons à d'autres protocoles.

- **Nombre de clusters**

Les Figures 4.10a et 4.10b montrent les nombres de clusters respectivement dans le réseau avec 100 nœuds et le réseau avec 500 nœuds. Nous constatons quand le

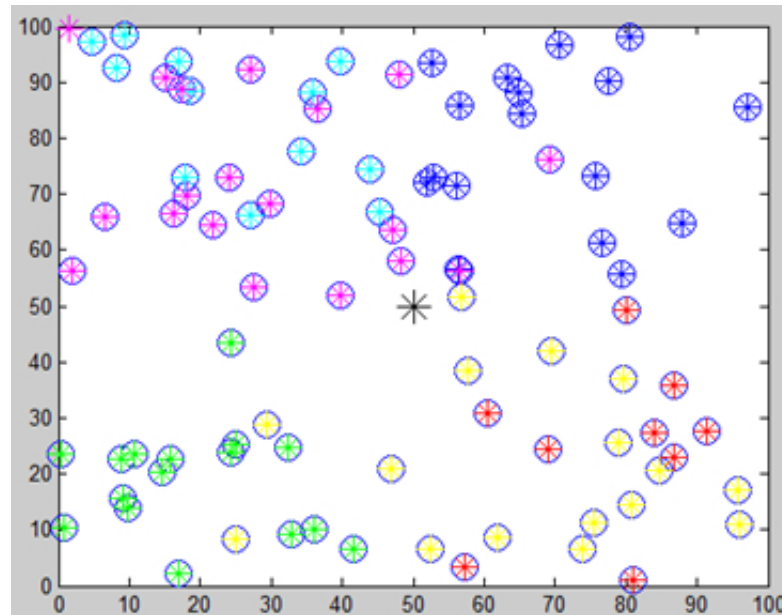


Figure 4.9: Clusters générés durant la phase de re-clustering

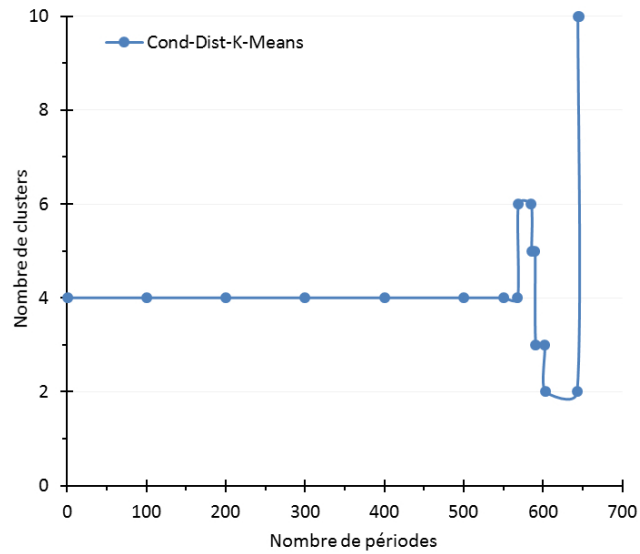
Table 4.2: Paramètres de simulation (Cond-Dist-K-Means)

Paramètres	Valeurs
Zone de déploiement	(0,0)x(100,100) et (0,0)x(200,200)
Position de la station de base	(50,50) et (100,100)
ϵ_{elec}	50 nJ/bit
ϵ_{friss_amp}	$10pJ/bit/m^2$
$\epsilon_{two_ray_amp}$	$0.0013pJ/bit/m^4$
d_0	87 m
Portée de transmission	30 m
Énergie initiale par capteur	0.5 J
Taille du réseau	100, 500
P	5%

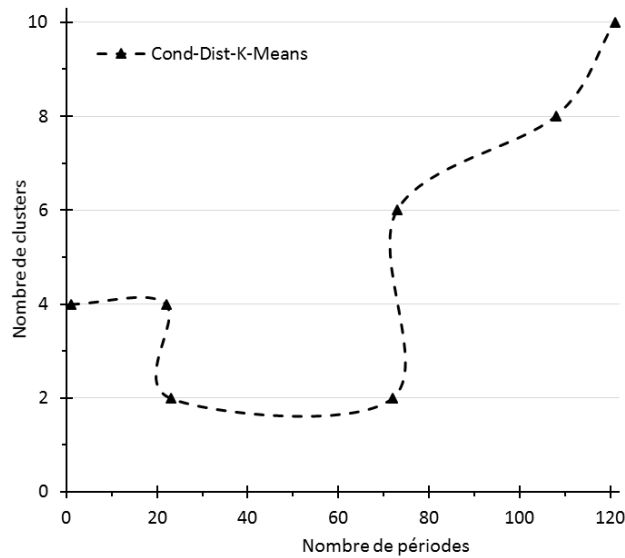
nombre de nœuds diminue le nombre de clusters augmente à cause de la dispersion des nœuds dans la zone de déploiement.

• Énergie Consommée

Les Figures 4.11a et 4.11b montrent la consommation d'énergie pendant la durée de vie du réseau. Le schéma de routage proposé offre de meilleurs résultats comparés au schéma de routage Dist-K-Means et au protocole LEACH en raison de la bonne répartition de la consommation de l'énergie pendant la durée de vie du réseau. De plus, dans notre contribution, le nombre de clusters formés en fonction du nombre des nœuds vivants a eu un impact positif sur les performances de notre contribution comparée à LEACH et à Dist-K-Means où le nombre de clusters est indépendant du nombre des nœuds vivants dans le réseau. Cependant, dans LEACH, le nombre de clusters est égal à 5 pour un réseau de 100 nœuds et 25 pour un réseau de 500



(a) Réseau contenant 100 noeuds



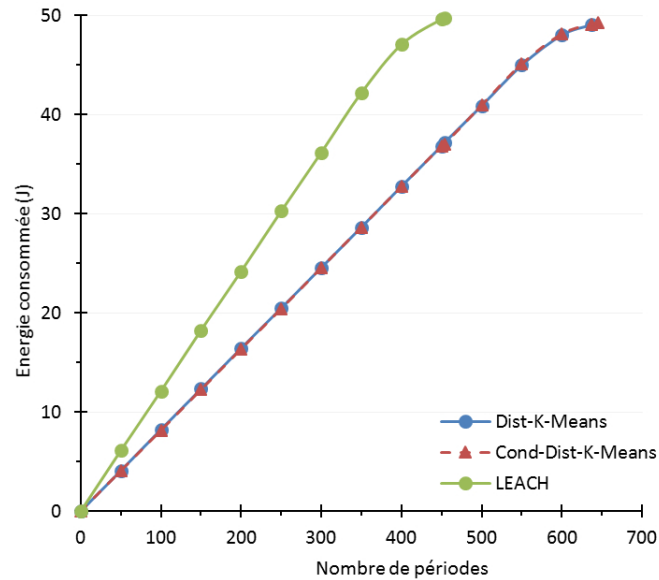
(b) Réseau contenant 500 noeuds

Figure 4.10: Nombre de clusters formés par Cond-Dist-K-Means

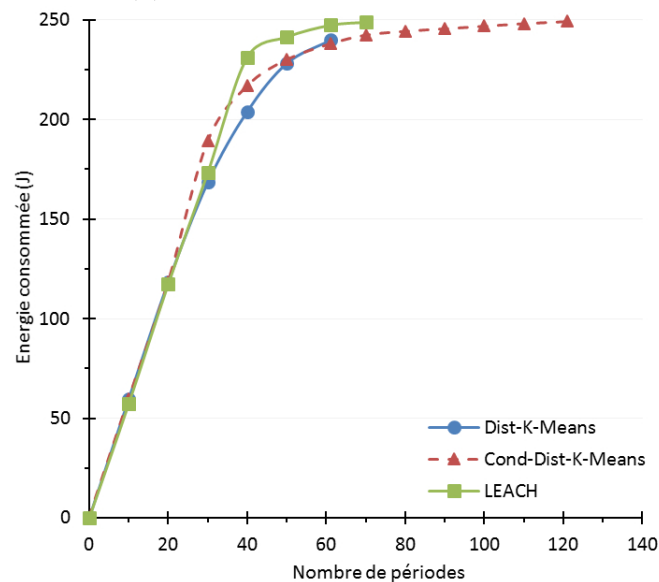
noeuds et dans le schéma de routage Dist-K-Means, il conserve toujours sa valeur initiale.

- **Durée de vie**

Les Figures 4.12a et 4.12b montrent le nombre de noeuds vivants au cours de la durée de vie du réseau. Nous supposons que lorsque le nombre de noeuds vivants atteignent 20 dans le premier réseau, ce dernier est considéré comme non-fonctionnel et dans le réseau avec 500 noeuds lorsque le nombre de noeuds vivants est moins de 100, nous considérons aussi qu'il n'est pas fonctionnel. Dans notre contribution, la restructuration du réseau en fonction du nombre de noeuds actifs a augmenté la durée de vie du réseau par rapport aux deux autres protocoles, comme illustré par la Figure 4.12.



(a) Réseau contenant 100 noeuds

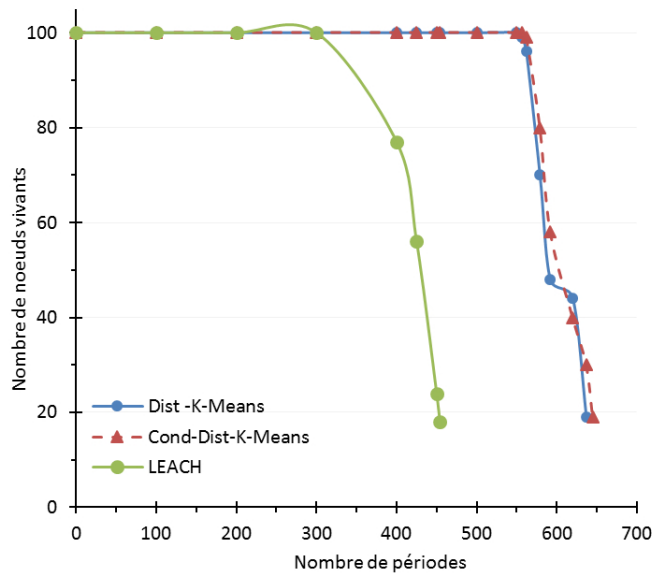


(b) Réseau contenant 500 noeuds

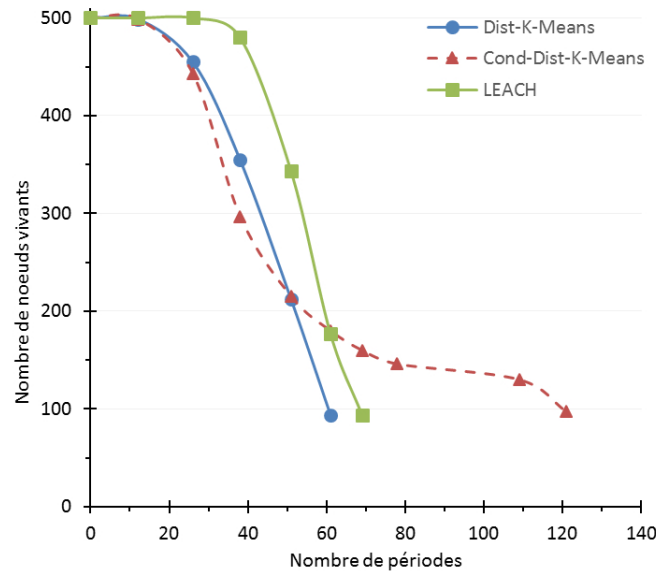
Figure 4.11: Énergie consommée en fonction du nombre de périodes

4.4.6 Contribution 3: Schéma de routage GA-Dist-K-Means

Dans cette contribution "GA-Dist-K-Means" [268], nous proposons un nouveau protocole de routage utilisant les AGs (les algorithmes génétiques) et K-Means pour prolonger la durée de vie des réseaux dans les RSCFs. Nous impliquons la méthode K-Means pour créer des clusters dans lesquels les CHs sont élus en fonction de la distance, de l'énergie résiduelle et du nombre de fois qu'un nœud était CH durant les périodes antérieures. De plus, nous utilisons une approche basée sur les AGs pour établir des chemins (CH-to-CH) d'un nœud source à la station de base. Le processus de clustering est effectué si tous les membres d'un même cluster sont morts et le processus d'établissement de chemins sera déclenché si un nœud change d'état, c'est-à-dire qu'un nœud membre devient un CH.



(a) Réseau contenant 100 noeuds



(b) Réseau contenant 500 noeuds

Figure 4.12: Nombre de noeuds vivants en fonction du nombre de périodes

a) Présentation de GA-Dist-K-Means

Dans ce schéma de routage, au début la station de base recueille les positions de tous les nœuds du réseau à l'aide d'un système de localisation. La solution de routage proposée divise le temps de simulation en périodes où chacune s'exécute en trois phases: la phase de clustering, la phase d'établissement de chemins et la phase de transmission des données comme le montre la Figure 4.13. Par ailleurs, lors de la première période, ces trois phases sont exécutées, mais dans les autres périodes, l'exécution de certaines phases dépend de certaines conditions, comme le montre l'organigramme de la Figure 4.13. En outre, nous supposons dans cette contribution que la station de base n'a pas de contrainte énergétique et qu'elle exécute les deux premières phases.

Le schéma de routage *GA-Dist-K-Means* est établi selon l'algorithme 4.

Algorithm 4 GA-Dist-K-Means

Require: (Network size, Node positions, Population size, Number of generations)

Ensure: Execution of GA-Dist-K-Means

```

1: if (round = 1) then
2:   Determination of clusters number
3:   Formation of clusters with K-Means
4:   Election of CHs in a centralized manner based an Energy, Average distance and
   number of times a node is elected as CH
5:   Construction of inter-cluster paths by Genetic Algorithm
6: end if
7: while (the network is functional) do
8:   if (All nodes in a cluster are dead) then
9:     Determination of clusters number
10:    Formation of clusters with K-Means
11:   end if
12:   if (A node changes its role) then
13:     Construction of inter-cluster paths by Genetic Algorithm
14:   end if
15:   Election of CHs in a centralized manner based en Energy, Average distance and
   Number of times a node is elected as CH
16:   The base station sends to each node in the network its affiliation and its next-hop
17:   Data transfer using multi-hop inter-cluster communication
18: end while

```

Phase de clustering

Cette phase comprend trois sous-phases: la sous-phase pour la détermination du nombre de clusters, la sous-phase de formation de clusters et la sous-phase d'élection des CHs. Les deux premières sous-phases sont exécutées si tous les membres d'un cluster sont morts tandis que la troisième sous-phase est exécutée après chaque round.

Au début la station de base détermine le nombre de clusters en utilisant la méthode de clustering "Rule of Thumb" [19]. Puis, elle lance la sous-phase de clustering durant laquelle des clusters seront formés en utilisant la méthode K-Means. Après ces deux sous-phases, la sous-phase d'élection des CHs aura lieu.

Lors de l'élection des CHs, la station de base choisit les nœuds qui ont plus d'aptitude pour ce rôle notamment ceux qui ont plus d'énergie. Le choix d'un noeud comme CH dépend de son poids qui est calculé en fonction de trois paramètres: l'énergie restante du nœud, la distance moyenne du nœud par rapport aux autres dans son cluster et le nombre de fois qu'un nœud était élu CH durant les périodes antérieures. Le poids d'un noeud est calculé selon l'équation 4.4 et la distance moyenne d'un nœud par rapport aux autres de son cluster est exprimée par l'équation 4.5.

$$Cost(Node_i) = \alpha * E(Node_i) + \frac{\beta}{AvgDist(Node_i)} + \frac{\gamma}{ec(Node_i)} \quad (4.4)$$

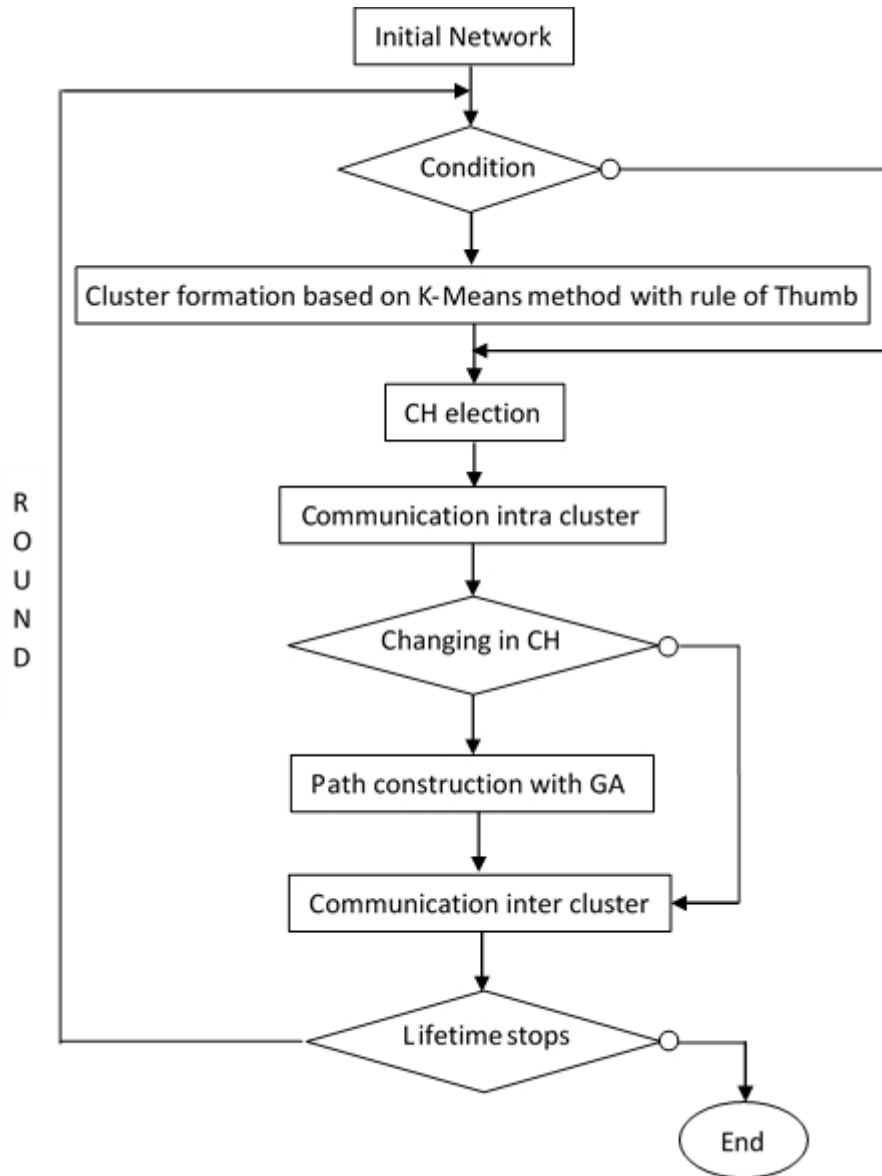


Figure 4.13: Organigramme de la démarche suivie

où $\alpha = 0.9$, $\beta = 0.05$ et $\gamma = 0.05$

où $ec(Node_i)$ représente le nombre de fois où $Node_i$ est élu CH et $E(Node_i)$ est l'énergie restante de $Node_i$. Nous favorisons la métrique énergie dans le processus d'élection des CHs, car ces derniers auront plusieurs tâches à accomplir et, par conséquent, ceux qui ont plus d'énergie seront les plus appropriés pour ce rôle.

$$AvgDist(Node_i) = \frac{\sum_{j=1}^{N_c} Dist(Node_i, Node_j)}{(N_c - 1)} \quad (4.5)$$

où N_c représente le nombre de noeuds dans le cluster.

Lorsqu'un nœud est élu comme CH, il devient responsable de plusieurs tâches telles

que la coordination entre les membres de son cluster, l'agrégation des données collectées (calcul de la moyenne) en un seul paquet et la transmission de ce paquet à la station de base, directement ou via d'autres CHs. Ces tâches exécutées plusieurs fois peuvent rapidement épuiser les batteries des CHs sous-jacents et l'épuisement de l'énergie de ces CHs peut entraîner des pannes qui affectent les performances et la fiabilité des RCSFs. En outre, si les CHs du réseau sont fixes, leurs batteries s'épuisent rapidement et cela provoque des trous dans le réseau. Pour cela, nous avons proposé que les CHs soient élus périodiquement après chaque période.

Si tous les membres d'un même cluster sont morts, le nombre de clusters sera déterminé une nouvelle fois par la station de base en faisant appel toujours à la méthode "Rule of Thumb" et de nouveaux clusters seront régénérés par la suite par la méthode K-Means.

Phase d'établissement des chemins

Dans le protocole proposé, la communication inter-cluster est effectuée de manière multi-sauts. Pour un RCSF avec N CHs, le nombre de chemins possibles d'un CH vers la station de base est exprimé selon l'équation 4.6. Ainsi, la sélection d'un chemin approprié dans les RCSFs est un problème NP-difficile et sa détermination dans un temps raisonnable nécessite une méthode d'optimisation (une méthode intelligente ou une métaheuristique). Une métaheuristique basée sur les AGs fonctionne bien pour résoudre le problème d'optimisation pour la recherche de chemins dans les RCSFs [276].

$$p(N) = 2^{(N+1)} - N + 1 \quad (4.6)$$

Dans cette phase, la recherche de chemins (CH-à-CH) entre chaque CH et la station de base est réalisée grâce à un algorithme génétique amélioré. Cet AG est exécuté si l'un des CHs change de statut et il prend comme paramètres d'entrée la position des CHs et leur énergie résiduelle. La Figure 4.14 représente le déroulement de la phase d'établissement de chemins.

Cette phase est exécutée durant la première période et si l'un des CHs ou un noeud membre change de statut. Dans ce qui suit, nous expliquons avec plus de détails le déroulement des étapes présentées dans l'organigramme de la Figure 4.14.

- **Population initiale et représentation des chromosomes**

Dans cette étape, le schéma de routage proposé définit la population initiale (chemins de routage initiaux). Nous notons que cette étape est critique pour le bon fonctionnement du schéma proposé car le bon choix d'une population initiale aura un impact positif sur la durée de la convergence de l'AG et sur la qualité des résultats. Néanmoins, la recherche d'une bonne population initiale est une problématique en elle-même. A cet effet, nous proposons que le choix de la population initiale se fait d'une manière aléatoire et qu'il suffit que cette population initiale contienne un ensemble de chemins valides. En outre, chaque individu de la population (chromosome) est représenté comme une chaîne contenant des numéros de nœuds (gènes). La longueur de chaque chromosome est égale au nombre de clusters où leurs CHs jouent le rôle de nœud relais.

Un exemple de schéma de routage contenant une station de base (BS) et six CHs est représenté par la Figure 4.15 et l'individu correspondant est représenté par la Figure 4.16. Dans cet exemple, la valeur 2 du gène en position 1, indique que le

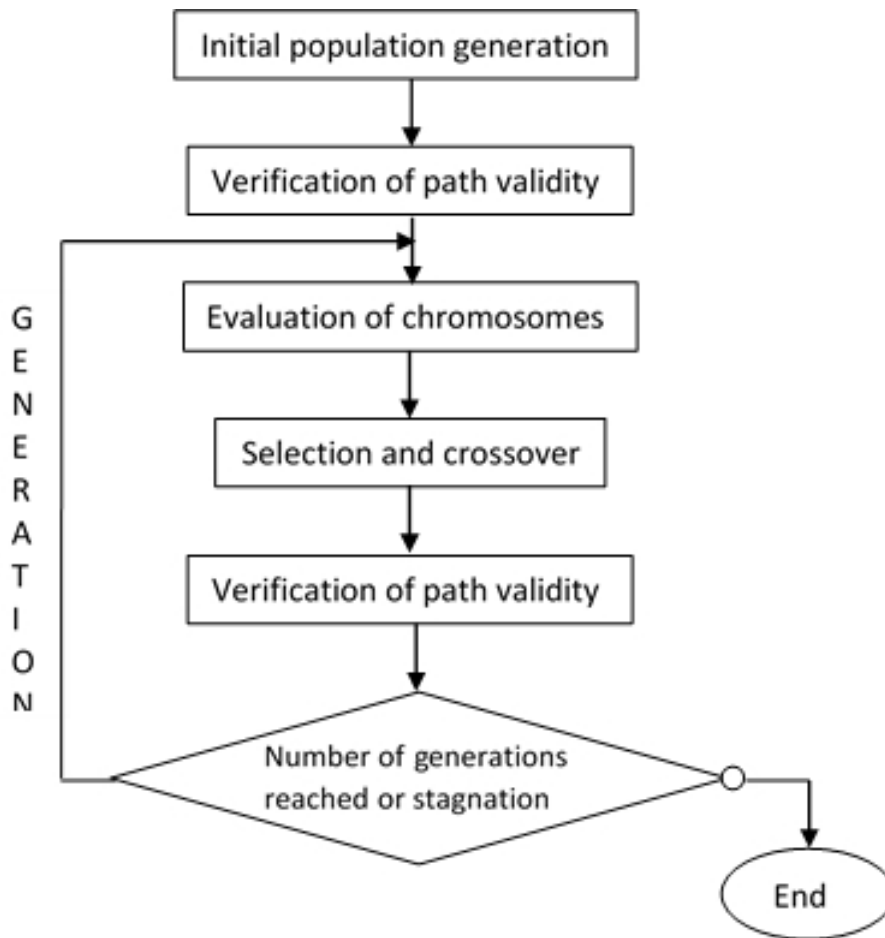


Figure 4.14: Organigramme de l'établissement des chemins en utilisant l'AG amélioré

nœud 1 transmet au nœud 2. De même, la valeur 7 du gène en position 3, signifie que le nœud 3 transmet au nœud 7 (BS).

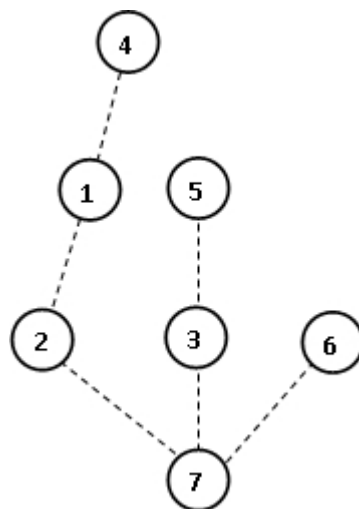


Figure 4.15: Sous-réseau formé de CHs et BS

1	2	3	4	5	6	From
2	7	7	1	3	7	To

Figure 4.16: Représentation d'un chromosome

- **Fonction objectif (Fitness)**

Dans l'AG, la fonction objectif est utilisée pour évaluer la qualité des individus (chromosomes), guider le processus évolutif et sélectionner la sortie finale. La conception d'une fonction Fitness adaptée est donc essentielle pour la performance de l'AG utilisé. Les facteurs permettant le choix de l'individu dans cette fonction Fitness comprennent l'énergie résiduelle, la distance et l'énergie requise pour envoyer les données. L'équation 4.7 représente l'évaluation de l'individu i .

$$ff(i) = \frac{\alpha * f_1(i) + \beta * f_2(i)}{f_3(i)} \quad (4.7)$$

où

$$f_1(i) = \frac{dist(i)}{\sum_{l=1}^p dist(l)}$$

$$f_2(i) = \frac{ensen(i)}{\sum_{l=1}^p ensen(l)}$$

$$f_3(i) = \sum_{j=1}^k en(l)$$

f_1 représente la somme des distances du chemin "i" par rapport à la distance globale qui sera couverte pour tous les chemins, f_2 est le rapport de l'énergie consommée par l'individu (chemin) i et l'ensemble des chemins dans cette population, et f_3 est l'énergie résiduelle globale. α et β sont les coefficients associés respectivement à l'énergie et à la distance dans la fonction Fitness, et $\alpha + \beta = 1$. Nous avons essayé d'équilibrer entre l'impact de l'énergie et l'impact de la distance en affectant à α la valeur 0,5 et à β la valeur 0,5. L'individu dont la valeur retournée par la fonction Fitness est la plus petite indique qu'il représente le chemin le plus approprié.

- **Sélection et croisement**

Lors de l'opération de sélection, les chromosomes qui ont les plus petites valeurs retournées par la fonction Fitness, sont sélectionnés pour la prochaine génération afin d'améliorer la qualité de la nouvelle population. Dans cette optique, la sélection élitiste et la sélection de tournois sont utilisées conjointement pour bénéficier de leurs avantages. Nous sélectionnons 50 % des individus qui ont des meilleures valeurs retournées par la fonction objectif pour la prochaine génération selon la stratégie élitiste et la deuxième partie de la population est générée en faisant appel à la stratégie de tournois.

Le croisement consiste à choisir deux chromosomes parents et d'échanger leurs informations génétiques pour générer deux fils et le processus de croisement peut prendre plusieurs formes. Dans le cadre de notre travail, le croisement en un point est appliqué, dans lequel la position du gène est choisie au hasard et les échanges de gènes se produisent à la position choisie sur le chromosome. Par exemple, considérons deux parents "2 7 7 1 3 7" et "3 7 7 1 6 7", et l'échange de gènes a eu lieu à la position 3, puis après croisement les deux chromosomes (les fils) sont "2 7 7 1 6 7" et "3 7 7 1 3 7". Après la phase de croisement, une autre étape est exécutée pour valider les chromosomes de la nouvelle population. Cette étape consiste à vérifier si chaque chromosome est valide c'est-à-dire pour chaque CH il y'a un chemin valide vers la station de base (BS). Après cette phase, la station de base informe par le biais d'un message ACK les nœuds capteurs de leurs nouvelles affiliations (à quels clusters ils appartiennent), de leurs status et de leurs nœuds relais.

Phase de transmission des données

Chaque nœud membre d'un cluster envoie directement les données collectées à son CH correspondant. Après la réception de tous les paquets par le CH correspondant, ce dernier les agrège en un seul paquet et le transmet à la station de base à travers un chemin (CH-to-CH) qui était établi durant la phase précédente.

b) Evaluation et résultats de simulations

Dans cette section, nous évaluons cette contribution qui consiste en un schéma de routage multi-sauts basé sur un clustering centralisé et qui implique un algorithme génétique amélioré. Puis, nous comparons ses performances au schéma de routage distribué Dist-K-Means [27] et au protocole LEACH [24] en termes de consommation d'énergie et de durée de vie des réseaux.

Environnement de simulations

Pour évaluer les performances de cette contribution, nous avons utilisé un réseau composé de 500 nœuds qui sont déployés d'une manière aléatoire dans une zone de 100 m x 100 m et la station de base est placée au point de coordonnées (120,50). Dans ces simulations, nous avons supposé que chaque nœud transmet un paquet de données par round à son CH correspondant et le nombre de clusters dans LEACH représente 10% du nombre total de nœuds du réseau ($P = 10\%$). Les paramètres utilisés dans les simulations sont résumés dans le tableau 4.3. Les simulations ont été exécutées 10 fois et les résultats fournis sont calculés sur la base de la moyenne de ces simulations.

Résultats de simulations

Dans la contribution proposée, la méthode de clustering "Rule of Thumb" est exécutée pour trouver le nombre optimal de clusters, l'approche K-Means pour le processus de clustering et l'AG pour le routage des données d'une manière différente que dans [149] où l'AG est utilisé pour trouver le nombre optimal de clusters et l'approche K-Means pour le processus de clustering. En outre, au cours d'une période, chaque membre du cluster envoie un seul paquet de données à son CH correspondant et à la fin de cette période, chaque CH agrège tous les paquets de données reçus de ses membres en un seul paquet

Table 4.3: Paramètres de simulation (GA-Dist-K-Means)

Paramètres	Valeurs
Zone de déploiement	(0,0)x(100,100)
Position de la station de base	(120,50)
ε_{elec}	50 nJ/bit
ε_{friss_amp}	10pJ/bit/m ²
$\varepsilon_{two_ray_amp}$	0.0013pJ/bit/m ⁴
d_0	87 m
Énergie initiale par noeud	2 J
Taille du réseau	500
P	10%

en calculant la moyenne et le transmet à la station de base directement ou en impliquant des CHs comme des noeuds relais.

- **Énergie Consommée**

La Figure 4.17 montre la consommation d'énergie pendant la durée de vie du réseau. Le schéma de routage proposé fournit de meilleurs résultats par rapport au schéma de routage Dist-K-Means et au protocole LEACH en raison du mode de transmission multi-sauts et également de la bonne répartition de la consommation d'énergie pendant la durée de vie du réseau. Nous notons que dans notre contribution la consommation d'énergie est plus importante jusqu'à la période 1200 par rapport aux autres protocoles. Cependant, à cette période dans les deux protocoles, les noeuds épuisent complètement leur énergie tandis que dans notre contribution les noeuds épuisent leur énergie à la période 1453 en raison de la bonne optimisation de la consommation d'énergie qui tient compte de son énergie résiduelle et de la distance qui sépare le noeud relais de la station de base.

- **Durée de vie**

La Figure 4.18 montre l'évolution du nombre de noeuds vivants pendant la durée de vie du réseau. Nous supposons que lorsque le nombre de noeuds vivants atteint 100, le réseau n'est pas considéré comme fonctionnel. GA-Dist-K-Means a permis de fournir une longue durée de vie du réseau par rapport aux deux autres protocoles comme illustré par la Figure 4.18 en raison de la bonne implication des noeuds comme des noeuds CHs et noeuds relais.

4.5 Récapitulatif sur les contributions proposées

Dans cette section, nous présentons un tableau récapitulatif dans lequel nous résumons les principales différences entre les contributions proposées et le protocole LEACH.

- Condition 1: quand le nombre des noeuds vivants diminue d'un certain seuil (5%), le processus de clustering sera relancé de nouveau.

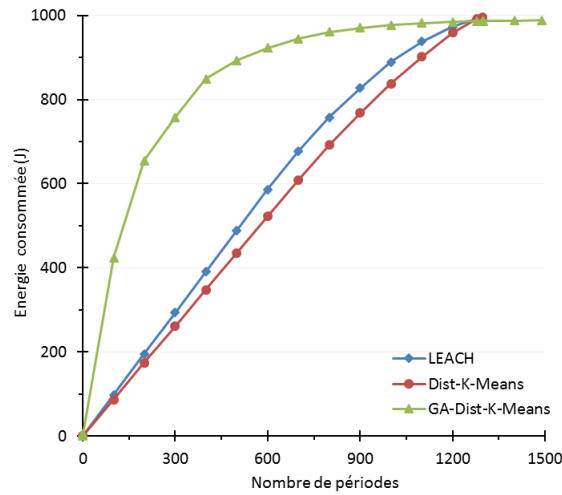


Figure 4.17: Énergie consommée vs. Nombre de périodes

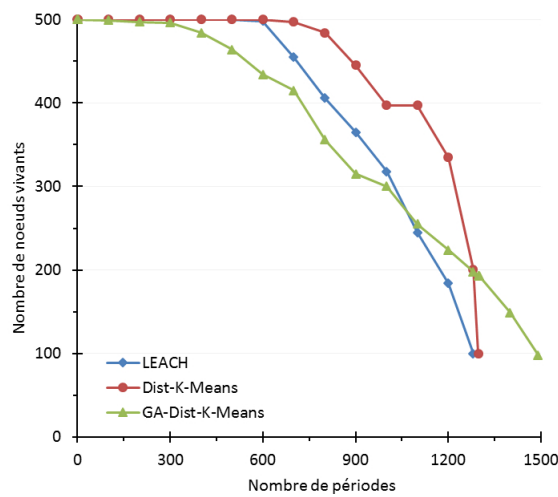


Figure 4.18: Nombre de noeuds vivants vs. Nombre de périodes

Table 4.4: Récapitulatif sur les contributions proposées

Protocole	Formation des clusters	Clustering	Nombre de clusters	Election des CHs	Communication Intercluster
LEACH [24]	Distribué	Dynamique (après chaque round)	Fixé	Aléatoire	Mono-saut
Dist-K-Means [27]	Centralisé	Statique	Fixé	Distance + Énergie	Mono-saut
Imp-Dist-K-Means [27]	Centralisé	Statique	Fixé	Distance + Énergie	Mono-saut
Cond-Dist-K-Means [28]	Centralisé	Variable (Condition 1)	Méthode Silhouette	Distance + Énergie	Mono-saut
GA-Dist-K-Means [268]	Centralisé	Variable (Condition 2)	Méthode "Rule of Thumb"	Distance + Énergie + Nombre de fois qu'il était CH	Multi-sauts

- Condition 2: quand tous les nœuds d'un même cluster meurent, le processus de clustering sera relancé de nouveau.

4.6 Conclusion

Dans ce chapitre, nous avons commencé par présenter un bref état de l'art sur les protocoles de routage basés sur le clustering dans les RCSFs, puis nous avons proposé trois contributions pour optimiser la consommation de l'énergie dans ce type réseaux. Pour ce faire, nous avons fait appel à plusieurs approches: la méthode K-Means, la méthode Silhouette, la méthode "Rule of Thumb" et les algorithmes génétiques.

Dans la première contribution, nous avons proposé deux schémas de routage basés sur l'approche K-means pour les RCSFs: Dist-K-Means et Imp-Dist-K-Means. Dans les deux schémas, l'élection des CHs est faite d'une manière distribuée. En outre, le deuxième schéma de routage proposé (Imp-Dist-K-Means) qui se base sur une version améliorée de K-Means permet de générer des clusters équilibrés. Par conséquent, la charge sera bien équilibrée entre les CHs élus.

Dans la deuxième contribution (Cond-Dist-K-Means), nous avons proposé un nouveau schéma de routage pour les RCSFs basé sur la méthode Silhouette pour trouver le nombre optimal de clusters et l'approche K-Means pour le processus de clustering. Cond-Dist-K-Means utilise un clustering dynamique où le processus de clustering sera déclenché une nouvelle fois quand le nombre des nœuds vivants descend au dessous d'un certain seuil (5 %) et l'élection des CHs est faite d'une manière distribuée.

Dans la troisième contribution (GA-Dist-K-Means), nous avons proposé un nouveau schéma de routage pour les RCSFs basé sur la méthode "Rule of Thumb" pour trouver le nombre optimal de clusters, l'approche K-Means pour le processus de clustering et un algorithme génétique amélioré pour le routage des données. Dans ce schéma de routage, une approche de clustering dynamique est proposée dans laquelle si tous les nœuds d'un cluster sont morts, le processus de clustering est déclenché de nouveau.

Les résultats de simulations ont montré que les schémas de routage proposés permettent de réduire la consommation d'énergie et prolonger la durée de vie des réseaux de capteurs par rapport au protocole LEACH en particulier pour les réseaux avec un grand nombre de nœuds (passage à l'échelle). En outre, le schéma de routage GA-Dist-K-Means a fourni de meilleurs performances comparé aux autres schémas de routage (Imp-Dist-K-Means et Dist-K-Means).

Conclusion Générale

Conclusion Générale

Les progrès récents dans le domaine des réseaux sans fil ont ouvert de nouveaux horizons dans plusieurs domaines d'applications (santé, bâtiment, agriculture, Transport, etc.). Dans cette thèse, nous nous sommes focalisés en particulier, sur le routage dans les RCSFs à grande échelle en faisant appel à une métaheuristique ou une méthode intelligente. Dans ce type de réseaux, la consommation d'énergie occupe une place exceptionnelle puisqu'elle se répercute sur la durée de vie des réseaux. Plusieurs travaux de recherche considèrent que l'efficacité énergétique est le facteur le plus important dans les protocoles de communication car dans les RCSFs, chaque nœud est alimenté par une puissance énergétique limitée qui est irremplaçable dans la majorité des cas. Par ailleurs, vue que ce type de réseaux nécessite dans certaines applications le déploiement d'un grand nombre de capteurs, l'efficacité énergétique et le passage à l'échelle sont des exigences clés dans les RCSFs. Afin de garantir ces exigences en même temps, un ensemble de protocoles de routage basés sur le concept de clustering a été développé dans le cadre de cette thèse.

Le clustering et le routage sont deux problèmes d'optimisation bien connus pour être des problèmes polynomiaux non déterministes (NP-difficiles) d'où une utilisation de méthodes intelligentes et/ou des métaheuristicues est nécessaire. Le clustering permet de structurer le réseau en blocs ou en clusters et de choisir pour chacun un nœud spécial appelé CH selon des critères bien définis. Ce nœud spécial assure la collecte d'informations des autres nœuds appartenant à son cluster correspondant et les envoie à travers des communications multi-sauts vers la station de base.

Bien que, le clustering est l'une des meilleures méthodes en termes de performance énergétique il présente quelques problèmes. Parmi ces problèmes, le problème de la détermination du nombre optimal de CHs et le problème de la détermination d'un arbre de communication inter-cluster qui permet de relier ces CHs avec la station de base.

Dans le cadre de cette thèse, nous avons proposé des solutions de routage semi distribuées basées sur des améliorations de l'approche K-Means pour les RCSFs dans le but de prolonger leurs durées de vie. Ces solutions sont résumées en quatre contributions.

Dans la première contribution, nous avons proposé une étude comparative sur les techniques les plus répandues permettant de déterminer le nombre de clusters approprié lors du processus de clustering par la méthode "Distributed K-Means". Ces techniques sont Rule of Thumb, Elbow, et Silhouette. Après la détermination du nombre de clusters par chacune des trois approches, le réseau sera clustérisé avec une version amélioré de la méthode K-means pour former les clusters. Puis, nous avons évalué les performances de chacune et les comparer entre elles et aux protocoles de routage LEACH, Imp_LEACH et MODLEACH. Nous avons impliqué trois scénarios avec des tailles de réseaux différentes. Le premier et le deuxième scénario concernent un réseau de taille moyenne alors que le troisième est un réseau dense contenant un grand nombre de nœuds pour évaluer la

robustesse de chaque méthode lors du passage à l'échelle. Les résultats de simulations ont montré que le schéma de routage basé sur la méthode "Rule of Thumb" offre de meilleures performances en termes de durée de vie des réseaux par rapport aux autres méthodes de clustering présentées dans cette thèse, en particulier lorsque la taille du réseau augmente.

Dans la deuxième contribution, nous avons proposé deux schémas de routage basés sur l'approche K-means pour les RCSFs "Dist-K-Means" et "Imp-Dist-K-Means". Les deux schémas proposés impliquent une méthode distribuée pour sélectionner les CHs. En outre, le second schéma proposé (*Imp-Dist-K-Means*) utilise une version améliorée de K-Means qui permet de générer des clusters équilibrés. Ceci est dans le but d'équilibrer la charge entre les CHs. Les résultats de simulations ont montré que les schémas de routage proposés permettent de réduire la consommation d'énergie et de prolonger la durée de vie des réseaux par rapport au protocole LEACH. De plus, les performances des deux schémas de routage proposés sont toujours meilleures, même pour les réseaux comportant un grand nombre de nœuds.

Dans la troisième contribution "Cond-Dist-K-Means", nous avons proposé un nouveau schéma de routage basé sur la méthode Silhouette pour trouver le nombre optimal de clusters et l'approche K-Means pour le processus de clustering. *Cond-Dist-K-Means* utilise un clustering dynamique après chaque diminution du pourcentage de nœuds actifs sous un seuil fixe (5%) et la sélection des CHs se fait de manière distribuée. Dans chaque cluster, les membres envoient leurs données à leurs CH correspondant qui, à son tour, regroupe les données en un seul paquet et l'envoie directement à la station de base. Les résultats de simulations ont montré que ce schéma de routage réduit la consommation d'énergie et prolonge la durée de vie des réseaux par rapport au protocole LEACH et à "Dist-K-Means" car la formation des clusters dépend du nombre de nœuds vivants et de leurs positions. De plus, les performances du schéma de routage proposé sont toujours meilleures même lorsque le nombre de nœuds dans le réseau augmente.

Dans la quatrième contribution "GA-Dist-K-Means", nous avons proposé un nouveau schéma de routage basé sur une hybridation de la méthode "Rule of Thumb" et de la méthode K-Means pour la formation des clusters et sur une version améliorée des algorithmes génétiques pour trouver les chemins intercluster. Le protocole de routage proposé utilise un clustering dynamique qui sera déclenché si tous les nœuds d'un même cluster épuisent leurs énergies. Pour cette contribution, les résultats de simulations ont montré que "GA-Dist-K-Means" permet de réduire la consommation d'énergie et de prolonger la durée de vie du réseau par rapport au protocole LEACH et à "Dist-K-Means".

Comme perspectives, nous comptons optimiser ces solutions de routage proposées en utilisant d'autres méthodes intelligentes afin pour préserver l'énergie dans les RCSFs à grande échelle et par suite augmenter leurs durées de vie.

Références Bibliographiques

Bibliography

- [1] G. Anastasi, M. Conti, M. D. Francesco, and A. Passarella, “Energy conservation in wireless sensor networks: A survey,” *Ad Hoc Networks*, vol. 7, no. 3, pp. 537–568, 2009.
- [2] P. Rawat, K. D. Singh, H. Chaouchi, and J. M. Bonnin, “Wireless sensor networks: a survey on recent developments and potential synergies,” *Journal of supercomputing*, vol. 68, no. 1, pp. 1–48, 2014.
- [3] H. Qin, Z. Zhang, R. Wang, X. Cai, and Z. Jia, “Energy-balanced and depth-controlled routing protocol for underwater wireless sensor networks,” in *International Conference on Algorithms and Architectures for Parallel Processing*, pp. 115–131, Springer, 2017.
- [4] I. F. Akyildiz and E. P. Stuntebeck, “Wireless underground sensor networks: Research challenges,” *Ad Hoc Networks*, vol. 4, no. 6, pp. 669–686, 2006.
- [5] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, “Wireless Multimedia Sensor Networks: Applications and Testbeds,” *Proceedings of the IEEE*, vol. 96, no. 10, pp. 1588–1605, 2008.
- [6] M. Salayma, A. Y. Al-Dubai, I. Romdhani, and Y. Nasser, “Wireless Body Area Network (WBAN): A Survey on Reliability, Fault Tolerance, and Technologies Co-existence,” *ACM Computing Surveys (CSUR)*, vol. 50, no. 1, pp. 1–38, 2017.
- [7] S. Lindsey and C. Raghavendra, “PEGASIS: Power-Efficient Gathering in Sensor Information Systems,” in *Proceedings of IEEE International Aerospace Conference*, vol. 3, pp. 1125–1130, 2002.
- [8] A. Manjeshwar and D. P. Agrawal, “TEEN: A Routing Protocol for Enhanced Efficiency in Wireless Sensor Networks,” in *Proceedings of the 15th International Symposium on Parallel and Distributed Processing*, pp. 2009–2015, April 2001.
- [9] Y. Xu, J. Heidemann, and D. Estrin, “Geography-informed energy conservation for ad hoc routing,” in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, pp. 70–84, ACM, 2001.
- [10] F. Soualah-Alila, *CAMLearn: Une Architecture de Système de Recommandation Sémantique Sensible au Contexte. Application au Domaine du M-Learning*. PhD thesis, Université de Bourgogne, France, March 2015.

- [11] H. El Alami and A. Najid, "Energy-efficient fuzzy logic cluster head selection in wireless sensor networks," in *Proceedings of IEEE International Conference on Information Technology for Organizations Development (IT4OD)*, pp. 1–7, 2016.
- [12] A. Merbah, *Méthodologie de conception d'une architecture sans fil : Application à la gestion des réseaux à grande échelle*. PhD thesis, Université Hassan II de Casablanca, Maroc, February 2014.
- [13] A. Awamry, D. Maga, and T. Nahdi, "Energy consumption of wireless sensor network," in *Proceedings of the 17th International Conference on Research in Telecommunication Technologies*, pp. 5–8, 2015.
- [14] B. Kechar, *Problématique de la consommation d'énergie dans les réseaux de capteurs sans fil*. PhD thesis, Thèse de Doctorat, Université d'Oran 1, July 2010.
- [15] V. Raghunathan, C. Schurgers, P. Sung, and M. Srivastava, "Energy-aware wireless microsensor networks," *Signal Processing Magazine, IEEE*, vol. 19, no. 2, pp. 40–50, 2002.
- [16] O. R. Merad Boudia, *Agrégation des données et sécurité des réseaux de capteurs sans fil*. PhD thesis, Université de Tlemcen, May 2014.
- [17] S. A. H. Belkhiria, S. Boukli Hacene, P. Lorenz, M. Belkheir, M. Gilg, and M. Bouziani, "WRE-OLSR, a new scheme for enhancing the lifetime within ad hoc and wireless sensor networks," *International Journal of Communication Systems*, vol. 32, no. 11, p. e3975, 2019.
- [18] M. B. Benmahdi and M. Lehsaini, "Performance evaluation of main approaches for determining optimal number of clusters in wireless sensor networks," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 33, no. 3, pp. 184–195, 2020.
- [19] T. Kodinariya and P. Makwana, "Review on determining number of cluster in K-means clustering," *International Journal of Advance Research in Computer Science and Management Studies*, vol. 1, no. 6, p. 90–95, 2013.
- [20] N. Bertagnolli, "Elbow method and finding the right number of clusters." Accessed December 2018.
- [21] K. Pollard and M. Van Der Laan, "A Method to Identify Significant Clusters in Gene Expression Data, Springer US," in *U.C. Berkeley Division of Biostatistics Working Paper Series*, pp. 21–40, 2012.
- [22] L. Kaufman and P. Rousseeuw, "Finding Groups in Data: An Introduction to Cluster Analysis," *John Wiley & Sons, Inc., Hoboken, New Jersey, USA*, 1990.
- [23] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press*, vol. 1, p. 281–297, 1967.

- [24] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless microsensor networks,” in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, vol. 2, pp. 1–10, January 2000.
- [25] O. Hassan, S. Krit, M. Kabrane, K. Bandaoud, M. Elaskri, K. Karimi, H. Elbousty, and L. Elmaimouni, “Energy Efficient in Wireless Sensor Networks Using Cluster-Based Approach Routing,” *International Journal of Sensors and Sensor Networks*, vol. 5, no. 5-1, pp. 6–12, 2017.
- [26] D. Mahmood, N. Javaid, S. Mahmood, S. Qureshi, A. M. Memon, and T. Zaman, “MODLEACH: a variant of LEACH for WSNs,” in *Proceedings of the 8th IEEE International Conference on Broadband and Wireless Computing, Communication and Applications*, pp. 158–163, IEEE, 2013.
- [27] M. Lehsaini and M. B. Benmahdi, “An Improved K-means cluster-based routing scheme for Wireless Sensor Networks,” in *Proceedings of the 13th International Symposium on Programming and Systems (ISPS’2018)*, pp. 1–6, April 2018.
- [28] M. B. Benmahdi and M. Lehsaini, “A Conditional Distributed K-means Cluster-based Routing Scheme for Wireless Sensor Networks,” in *Proceedings of the Second International Conference on Artificial Intelligence and Information Technology, ICA2IT’19*, pp. 1–6, April 2019.
- [29] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: a survey,” *Computer networks*, vol. 38, no. 4, pp. 393–422, 2002.
- [30] T. Rault, *Energy-efficiency in wireless sensor networks*. PhD thesis, Université de Technologie de Compilègne, October 2015.
- [31] C. Boano, J. Brown, Z. He, U. Roedig, and T. Voigt, “Low-Power radio communication in industrial outdoor deployments: The impact of weather conditions and ATEX-Compliance,” in *Sensor Applications, Experimentation, and Logistics* (N. Komninos, ed.), vol. 29 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 159–176, Springer Berlin Heidelberg, 2010.
- [32] C. Mettu, “Telosb datasheet @ONLINE.” Accessed October 2018.
- [33] K. Lin, J. Yu, J. Hsu, S. Zahedi, D. Lee, J. Friedman, A. Kansal, V. Raghunathan, and M. Srivastava, “Helimote: Enabling Long-lived Sensor Networks Through Solar Energy Harvesting,” in *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems (SenSys’05)*, pp. 309–309, ACM, 2005.
- [34] T. Voigt, H. Ritter, and J. Schiller, “Utilizing solar power in wireless sensor networks,” in *Proceedings of the 28th Annual IEEE International Conference on Local Computer Networks (LCN’03)*, pp. 416–422, October 2003.
- [35] S. Harchi, *Un protocole de session dans les réseaux de capteurs sans fil*. PhD thesis, Université de Lorraine, France, 2013.

- [36] B. Malli, “Micaz datasheet @ONLINE.” Accessed May 2018.
- [37] Q. Mamun, “A Qualitative Comparison of Different Logical Topologies for Wireless Sensor Networks,” *Sensors (Basel)*, vol. 12, no. 11, pp. 14887–14913, 2012.
- [38] D. Xu and J. Gao, “Comparison study to hierarchical routing protocols in wireless sensor networks ,” *Procedia Environmental Sciences*, vol. 10, p. 595–600, 2011.
- [39] P. S. Mann and S. Singh, “ Energy-efficient hierarchical routing for wireless sensor networks: a swarm intelligence approach ,” *Wireless Personal Communications*, vol. 92, no. 2, p. 785–805, 2017.
- [40] O. N. Samijayani, H. Firdaus, and A. Mujadin, “Solar energy harvesting for wireless sensor networks node,” in *Proceedings of International Symposium on Electronics and Smart Devices (ISESD)*, pp. 30–33, October 2017.
- [41] S. M. Mazinani, H. Yousefi, and M. Mirzaie, “ A vector-based routing protocol in underwater wireless sensor networks,” *Wireless Personal Communications*, vol. 100, no. 4, pp. 1569–1583, 2018.
- [42] M. Ahmed, M. Salleh, and M. Channa, “Routing protocols based on protocol operations for underwater wireless sensor network: A survey,” *Egyptian Informatics Journal*, vol. 19, no. 1, pp. 57–62, 2018.
- [43] A. Yahya, S. U. Islam, A. Akhuzada, G. Ahmed, S. Shamshirband, and J. Lloret, “Towards Efficient Sink Mobility in Underwater Wireless Sensor Networks,” *Energies*, vol. 11, no. 6, pp. 1–12, 2018.
- [44] J. Burrell, T. Brooke, and R. Beckwith, “Vineyard computing: sensor networks in agricultural production,” *IEEE Pervasive Computing*, vol. 3, no. 1, pp. 38–45, 2004.
- [45] R. Cardell-Oliver, K. Smettem, M. Kranz, and K. Mayer, “A reactive soil moisture sensor network: design and field evaluation,” *International Journal of Distributed Sensor Networks*, vol. 1, no. 2, p. 149–162, 2005.
- [46] S. Cheekiralla, *Development of a wireless sensor unit for tunnel monitoring*. PhD thesis, Massachusetts Institute of Technology, 2004.
- [47] G. E. Rolader, J. Rogers, and J. Batteh, “Self-healing minefield,” in *Battlespace Digitization and Network-Centric Systems IV* (R. Suresh, ed.), vol. 5441, pp. 13–24, International Society for Optics and Photonics, SPIE, 2004.
- [48] D. J. Daniels, “Surface-penetrating radar,” *Electronics Communication Engineering Journal*, vol. 8, no. 4, pp. 165–182, 1996.
- [49] T. B. Leyser and A. Y. Wong, “Powerful electromagnetic waves for active environmental research in geospace,” *Reviews of Geophysics*, vol. 47, no. 1, pp. 1–33, 2009.
- [50] S. A. Alvi, B. Afzal, G. A. Shah, L. Atzori, and W. Mahmood, “Internet of multi-media things: Vision and challenges,” *Ad Hoc Networks*, vol. 33, pp. 87–111, 2015.

-
- [51] J. N. Al-Karaki and A. Gawanmeh, “The Optimal Deployment, Coverage, and Connectivity Problems in Wireless Sensor Networks: Revisited,” *IEEE Access*, vol. 5, pp. 18051–18065, 2017.
- [52] W. Itani, A. Kayssi, and A. Chehab, “Wireless body sensor networks: Security, privacy, and energy efficiency in the era of cloud computing,” *International Journal of Reliable and Quality E-Healthcare*, vol. 5, no. 2, pp. 1–30, 2016.
- [53] R. Miramontes, R. Aquino, A. Flores, G. Rodríguez, R. Anguiano, A. Ríos, and A. Edwards, “PlaIMoS: A Remote Mobile Healthcare Platform to Monitor Cardiovascular and Respiratory Variables,” *Sensors (Basel)*, vol. 17, no. 1, pp. 1–24, 2017.
- [54] C. C. Shen, C. Srisathapornphat, and C. Jaikaeo, “Sensor information networking architecture and applications,” *IEEE Personal Communications*, vol. 8, no. 4, pp. 52–59, 2001.
- [55] G. Hoblos, M. Staroswiecki, and A. Aitouche, “Optimal design of fault tolerant sensor networks,” in *Proceedings of IEEE International Conference on Control Applications*, pp. 467–472, September 2000.
- [56] N. Bulusu, D. Estrin, L. Girod, and J. Heidemann, “Scalable Coordination for Wireless Sensor Networks: Self-Configuring Localization Systems,” in *Proceedings of the 6th IEEE International Symposium on Communication Theory and Application*, (St. Martin’s College, Ambleside, Cumbria, UK), pp. 1–6, IEEE, July 2001.
- [57] E. Shih, S. Cho, N. Ickes, R. Min, A. Sinha, A. Wang, and A. Chandrakasan, “Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks,” in *Proceedings of ACM MobiCom’01*, p. 272–286, July 2001.
- [58] A. E. Cerpa, J. E. Elson, D. Estrin, L. D. Girod, M. Hamilton, and J. Zhao, “Habitat monitoring: application driver for wireless communications technology,” *ACM SIGCOMM Computer Communication Review*, vol. 31, no. 2, p. 20–41, 2001.
- [59] J. M. Rabaey, M. J. Ammer, J. L. da Silva, D. Patel, and S. Roundy, “Picoradio supports ad hoc ultra-low power wireless networking,” *Computer*, vol. 33, no. 7, pp. 42–48, 2000.
- [60] J. Rabaey, J. Ammer, J. L. Da Silva, and D. Patel, “PicoRadio: Ad-hoc wireless networking of ubiquitous low-energy sensor/monitor nodes,” in *Proceedings IEEE Computer Society Workshop on VLSI 2000. System Design for a System-on-Chip Era*, pp. 9–12, April 2000.
- [61] C. Intanagonwiwat, R. Govindan, and D. Estrin, “Directed diffusion: a scalable and robust communication paradigm for sensor networks,” in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom’00)*, 2000.
- [62] G. J. Pottie and W. J. Kaiser, “Wireless integrated network sensors,” *Communications of the ACM*, vol. 43, no. 5, p. 551–558, 2000.

- [63] J. Kahn, R. Katz, and K. Pister, "Next century challenges: mobile networking for smart dust," in *Proceedings of the ACM MobiCom'99*, p. 271–278, September 1999.
- [64] A. Porret, T. Melly, C. C. Enz, and E. A. Vittoz, "A low-power low-voltage transceiver architecture suitable for wireless distributed sensors network," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, vol. 1, pp. 56–59, May 2000.
- [65] T. Melly, A. S. Porret, C. Enz, M. Kayal, and E. Vittoz, "A 1.2V, 430MHz, 4dBm power amplifier and a 250 μ W front-end, using a standard digital CMOS process," in *Proceedings of the International symposium on Low power electronics and design (ISLPED '99)*, p. 233–237, August 1999.
- [66] A. Woo and D. Culler, "A transmission control scheme for media access in sensor networks," in *Proceedings of ACM MobiCom'01*, p. 221–235, July 2001.
- [67] S. Ullah, H. Higgins, B. Braem, B. Latre, C. Blondia, I. Moerman, S. Saleem, Z. Rahman, and K. S. Kwak, "A Comprehensive Survey of Wireless Body Area Networks," *Journal of Medical Systems*, vol. 36, p. 1065–1094, June 2012.
- [68] R. Latif, H. Abbas, and S. Assar, "Distributed denial of service (DDoS) attack in cloud- assisted wireless body area networks: a systematic literature review," *Journal of Medical Systems*, vol. 38, no. 128, pp. 1–10, 2012.
- [69] M. A. Jan, P. Nanda, X. He, and R. P. Liu, "A sybil attack detection scheme for a forest wildfire monitoring application," *Future Generation Computer Systems*, vol. 80, pp. 613 – 626, 2018.
- [70] D. Puthal, S. Nepal, R. Ranjan, and J. Chen, "DPBSV - An Efficient and Secure Scheme for Big Sensing Data Stream," in *Proceedings of IEEE Trustcom/BigDataSE/ISPA*, vol. 1, pp. 246–253, August 2015.
- [71] J. Mian, N. Priyadarsi, U. Muhammad, and H. Xiangjian, "PAWN: A payload-based mutual authentication scheme for wireless sensor networks," *Concurrency Computation: Practice and Experience*, vol. 29, no. 17, pp. 1–10, 2017.
- [72] D. Puthal, B. Sahoo, and B. P. S. Sahoo, "Effective Machine to Machine Communications in Smart Grid Networks," *ARPJ Journal of Systems and Software*, vol. 2, no. 1, pp. 18–22, 2012.
- [73] M. Usman, M. A. Jan, and X. He, "Cryptography-based secure data storage and sharing using hevc and public clouds," *Information Sciences*, vol. 387, pp. 90–102, 2017.
- [74] S. Khan, F. Khan, F. Arif, Q. Jabeen, and M. A. Jan, "Performance Improvement in Wireless Sensor and Actor Networks," *Journal of Applied Environmental and Biological Sciences*, vol. 6, no. 4S, pp. 191–200, 2016.
- [75] M. Usman, M. A. Jan, X. He, and P. Nanda, "Data Sharing in Secure Multimedia Wireless Sensor Networks," in *Proceedings of IEEE Trustcom/BigDataSE/ISPA*, pp. 590–597, August 2016.

- [76] F. Khan, F. Bashir, and K. Nakagawa, "Dual head clustering scheme in wireless sensor networks," in *Proceedings of International Conference on Emerging Technologies*, pp. 1–5, October 2012.
- [77] E. Sardini and M. Serpelloni, "Self-Powered Wireless Sensor for Air Temperature and Velocity Measurements With Energy Harvesting Capability," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, pp. 1838–1844, May 2011.
- [78] M. Magno, T. Polonelli, L. Benini, and E. Popovici, "A Low Cost, Highly Scalable Wireless Sensor Network Solution to Achieve Smart LED Light Control for Green Buildings," *IEEE Sensors Journal*, vol. 15, no. 5, pp. 2963–2973, 2015.
- [79] N. Ahmed, D. De, and I. Hussain, "Internet of Things (IoT) for Smart Precision Agriculture and Farming in Rural Areas," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4890–4899, 2018.
- [80] J. J. Estrada-López, A. A. Castillo-Atoche, J. Vázquez-Castillo, and E. Sánchez-Sinencio, "Smart Soil Parameters Estimation System Using an Autonomous Wireless Sensor Network With Dynamic Power Management Strategy," *IEEE Sensors Journal*, vol. 18, no. 21, pp. 8913–8923, 2018.
- [81] S. N. Daskalakis, G. Goussetis, S. D. Assimonis, M. M. Tentzeris, and A. Georgiadis, "A uW Backscatter-Morse-Leaf Sensor for Low-Power Agricultural Wireless Sensor Networks," *IEEE Sensors Journal*, vol. 18, no. 19, pp. 7889–7898, 2018.
- [82] C. Alippi, R. Camplani, C. Galperti, and M. Roveri, "A Robust, Adaptive, Solar-Powered WSN Framework for Aquatic Environmental Monitoring," *IEEE Sensors Journal*, vol. 11, no. 1, pp. 45–55, 2011.
- [83] F. Ingelrest, G. Barrenetxea, G. Schaefer, M. Vetterli, O. Couach, and M. Parlange, "SensorScope: Application-specific sensor network for environmental monitoring," *ACM Transactions on Sensor Networks (TOSN)*, vol. 6, p. Article 17, March 2010.
- [84] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pp. 88–97, 2002.
- [85] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet," *ACM SIGARCH Computer Architecture News*, vol. 30, no. 5, pp. 96–107, 2002.
- [86] L. Muduli, D. P. Mishra, and P. K. Jana, "Application of wireless sensor network for environmental monitoring in underground coal mines: a systematic review," *Journal of Network and Computer Applications*, vol. 106, pp. 48–67, 2018.
- [87] Y. Liu, Y. Liu, H. Xu, and K. L. Teo, "Forest fire monitoring, detection and decision making systems by wireless sensor network," in *Proceedings of IEEE Chinese Control And Decision Conference (CCDC)*, pp. 5482–5486, 2018.

- [88] A. Huang, M. Huang, Z. Shao, X. Zhang, D. Wu, and C. Cao, “A Practical Marine Wireless Sensor Network Monitoring System Based on LoRa and MQTT,” *arXiv preprint arXiv:1906.09571*, 2019.
- [89] V. Bapat, P. Kale, V. Shinde, N. Deshpande, and A. Shaligram, “Wsn application for crop protection to divert animal intrusions in the agricultural land,” *Computers and electronics in agriculture*, vol. 133, pp. 88–96, 2017.
- [90] F. Mohamedali and N. Matorian, “Support Dementia: Using Wearable Assistive Technology and Analysing Real-Time Data,” in *Proceedings of International Conference on Interactive Technologies and Games (ITAG)*, pp. 50–54, October 2016.
- [91] M. Chan, D. Estève, C. Escriba, and E. Campo, “A review of smart homes: Present state and future challenges,” *Computer Methods and Programs in Biomedicine*, vol. 91, no. 1, pp. 55–81, 2008.
- [92] P. Kakria, N. Tripathi, and P. Kitipawang, “A Real-Time Health Monitoring System for Remote Cardiac Patients Using Smartphone and Wearable Sensors,” *International journal of telemedicine and applications*, vol. 2015, pp. 1–11, 2015.
- [93] J. Chen, S. Kher, and A. Somani, “Distributed Fault Detection of Wireless Sensor Networks,” in *Proceedings of the Workshop on Dependability Issues in Wireless Ad Hoc Networks and Sensor Networks*, pp. 65–72, ACM, 2006.
- [94] N. S. Ali and Z. A. A. Alyasseri, “Wireless Sensor Network and Web Application Hybrid Scheme for Healthcare Monitoring,” *Journal of Soft Computing and Decision Support Systems*, vol. 4, no. 5, pp. 1–7, 2017.
- [95] E. Petriu, N. D. Georganas, D. Petriu, D. Makrakis, and V. Groza, “Sensor-based information appliances,” *Instrumentation Measurement Magazine, IEEE*, vol. 3, no. 4, pp. 31–35, 2000.
- [96] V. Ricquebourg, D. Menga, D. Durand, B. Marhic, L. Delahoche, and C. Loge, “The smart home concept: our immediate future,” in *Proceedings of the 1st IEEE International Conference on E-Learning in Industrial Electronics*, pp. 23–28, December 2006.
- [97] A. M. Tabar, A. Keshavarz, and H. Aghajan, “Smart home care network using sensor fusion and distributed vision-based reasoning,” in *Proceedings of the 4th ACM International Workshop on Video Surveillance and Sensor Networks*, pp. 145–154, ACM, 2006.
- [98] D.-M. Han and J.-H. Lim, “Smart home energy management system using IEEE 802.15.4 and Zigbee,” *IEEE Transactions on Consumer Electronics*, vol. 56, no. 3, pp. 1403–1410, 2010.
- [99] Y. Zhang and H. Xiao, “Bluetooth-Based Sensor Networks for Remotely Monitoring the Physiological Signals of a Patient,” *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 6, pp. 1040–1048, 2009.

-
- [100] C. M. Yu and Y. B. Yu, “Reconfigurable Algorithm for Bluetooth Sensor Networks,” *IEEE Sensors Journal*, vol. 14, no. 10, pp. 3506–3507, 2014.
- [101] R. de Francisco, H. Li, G. Dolmans, and H. de Groot, “Coexistence of ZigBee wireless sensor networks and Bluetooth inside a vehicle,” in *Proceedings of the 20th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 2700–2704, September 2009.
- [102] L. Hwakyung, C. Sang-Hwa, L. Yun-Sung, and H. Yoobin, “Performance Comparison of DASH7 and ISO/IEC 18000-7 for Fast Tag Collection with an Enhanced CSMA/CA Protocol,” in *Proceedings of the 10th IEEE International Conference on Embedded and Ubiquitous Computing (HPCC_EUC), and High Performance Computing and Communications*, pp. 769–776, November 2013.
- [103] H. Hongwei, X. Youzhi, Y. Hairong, S. Mubeen, and Z. Hongke, “An Elderly Health Care System Using Wireless Sensor Networks at Home,” in *Proceedings of the 3th International Conference on Sensor Technologies and Applications (SENSORCOMM’09)*, pp. 158–163, June 2009.
- [104] P. Kuila, S. K. Gupta, and P. K. Jana, “A novel evolutionary approach for load balanced clustering problem for wireless sensor networks,” *Swarm and Evolutionary Computation*, vol. 12, pp. 48–56, 2013.
- [105] A. Mahboub, M. Arioua, E. M. En-Naimi, and I. Ez-zazi, “Performance Evaluation of Energy-efficient Clustering Algorithms in Wireless Sensor Networks,” *Journal of Theoretical & Applied Information Technology*, vol. 83, no. 2, 2016.
- [106] N. Lasla, “La gestion de clés dans les réseaux de capteurs sans-fil,” *Mémoire de Magister, Institut National de formation en Informatique (INI), Oued-Smar, Alger*, 2007.
- [107] D. Hamdan, *Détection et diagnostic des fautes dans des systèmes à base de réseaux de capteurs sans fils*. PhD thesis, Grenoble, 2013.
- [108] A. A. Abbasi and M. Younis, “A survey on clustering algorithms for wireless sensor networks,” *Computer Communications*, vol. 30, no. 14, pp. 2826–2841, 2007.
- [109] M. Sajwan, A. K. Sharma, and K. Verma, “Analysis of scalability for hierarchical routing protocols in wireless sensor networks,” in *Proceedings of ICETIT 2019*, pp. 107–116, Springer, 2020.
- [110] S. M. Hedetniemi, S. T. Hedetniemi, and A. L. Liestman, “A survey of gossiping and broadcasting in communication networks,” *Networks*, vol. 18, no. 4, pp. 319–349, 1988.
- [111] W. R. Heinzelman, J. Kulik, and H. Balakrishnan, “Adaptive protocols for information dissemination in wireless sensor networks,” in *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile computing and networking (MobiCom ’99)*, pp. 174–185, August 1999.

- [112] J. Kulik, W. Heinzelman, and H. Balakrishnan, "Negotiation-based Protocols for Disseminating Information in Wireless Sensor Networks," *Wireless Networks*, vol. 8, pp. 169–185, March 2002.
- [113] K. Sohrabi, J. Gao, V. Ailawadhi, and G. J. Pottie, "Protocols for self-organization of a wireless sensor network," *IEEE Personal Communications*, vol. 7, no. 5, pp. 16–27, 2000.
- [114] M. Chu, H. Haussecker, and F. Zhao, "Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks," *International Journal of High Performance Computing Applications*, vol. 16, no. 3, pp. 293–313, 2002.
- [115] F. Zabin, S. Misra, I. Woungang, H. F. Rashvand, N.-W. Ma, and M. A. Ali, "REEP: data-centric, energy-efficient and reliable routing protocol for wireless sensor networks," *IET Communications*, vol. 2, no. 8, pp. 995–1008, 2008.
- [116] N. V. K. Subramanian, "Survey on Energy-Aware Routing and Routing Protocols for Sensor Networks," tech. rep., Technical Report, Department of Computer Science University of North Carolina Charlotte, North Carolina, USA, 2004.
- [117] M. Sabet and H. Naji, "An energy efficient multi-level route-aware clustering algorithm for wireless sensor networks: A self-organized approach," *Computers & Electrical Engineering*, vol. 56, pp. 399–417, 2016.
- [118] P. Kuila and P. K. Jana, "An energy balanced distributed clustering and routing algorithm for Wireless Sensor Networks," in *Proceedings of 2nd IEEE International Conference on Parallel, Distributed and Grid Computing*, pp. 220–225, IEEE, 2012.
- [119] A. Chamam and S. Pierre, "A distributed energy-efficient clustering protocol for wireless sensor networks," *Computers & Electrical Engineering*, vol. 36, no. 2, pp. 303–312, 2010.
- [120] M. Sabet and H. R. Naji, "A decentralized energy efficient hierarchical cluster-based routing algorithm for wireless sensor networks," *AEU-International Journal of Electronics and Communications*, vol. 69, no. 5, pp. 790–799, 2015.
- [121] M. Sajwan, D. Gosain, and A. K. Sharma, "Hybrid energy-efficient multi-path routing for wireless sensor networks," *Computers & Electrical Engineering*, vol. 67, pp. 96–113, 2018.
- [122] K. T. Kim, C. H. Lyu, S. S. Moon, and H. Y. Youn, "Tree-based clustering (TBC) for energy efficient wireless sensor networks," in *Proceedings of 24th IEEE International Conference on Advanced Information Networking and Applications Workshops*, pp. 680–685, 2010.
- [123] M. Sajwan, D. Gosain, and A. K. Sharma, "CAMP: cluster aided multi-path routing protocol for wireless sensor networks," *Wireless Networks*, vol. 25, no. 5, pp. 2603–2620, 2019.

- [124] R. D. Gawade and S. L. Nalbalwar, “A Centralized Energy Efficient Distance Based Routing Protocol for Wireless Sensor Networks,” *Journal of Sensors*, vol. 2016, pp. 1–8, 2016.
- [125] S. Sharma and S. K. Jena, “Cluster Based Multipath Routing Protocol for Wireless Sensor Networks,” *ACM SIGCOMM Computer Communication Review*, vol. 45, no. 2, pp. 14–20, 2015.
- [126] A. Kumar, H. Y. Shwe, K. J. Wong, and P. H. Chong, “Location-based routing protocols for wireless sensor networks: A survey,” *Wireless Sensor Network*, vol. 9, no. 1, pp. 25–72, 2017.
- [127] Y. Yu, R. Govindan, and D. Estrin, “Geographical and energy aware routing: A recursive data dissemination protocol for wireless sensor networks,” tech. rep., Technical report ucla/csd-tr-01-0023, UCLA Computer Science Department, 2001.
- [128] B. Karp and H. T. Kung, “GPSR: Greedy Perimeter Stateless Routing for Wireless Networks,” in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, MobiCom ’00, pp. 243–254, ACM, 2000.
- [129] E. Sammut and C. J. Debono, “A location-based routing algorithm for wireless sensor networks,” in *Proceedings of IEEE International Conference on Computer as a Tool (EUROCON)*, pp. 1–5, 2015.
- [130] J. Lou, “Location-Based Multi-objective Optimization Routing Algorithm for WSN,” in *Recent Developments in Intelligent Computing, Communication and Devices*, pp. 523–529, Springer, 2019.
- [131] I. H. Osman and J. P. Kelly, “Meta-heuristics theory and applications,” *Journal of the Operational Research Society*, vol. 48, no. 6, pp. 657–657, 1997.
- [132] J.-K. Hao, P. Galinier, and M. Habib, “Métaheuristiques pour l’optimisation combinatoire et l’affectation sous contraintes,” *Revue d’intelligence artificielle*, vol. 13, no. 2, pp. 283–324, 1999.
- [133] R. V. Kulkarni, A. Förster, and G. K. Venayagamoorthy, “Computational Intelligence in Wireless Sensor Networks: A Survey,” *IEEE Communications Surveys Tutorials*, vol. 13, no. 1, pp. 68–96, 2011.
- [134] S. Hussain, A. W. Matin, and O. Islam, “Genetic Algorithm for Hierarchical Wireless Sensor Networks,” *Journal of Networks*, vol. 2, no. 5, pp. 87–97, 2007.
- [135] A. W. Matin and S. Hussain, “Intelligent hierarchical cluster-based routing,” in *Proceedings of International Workshop on Mobility and Scalability in Wireless Sensor Networks (MSWSN) in IEEE International Conference on Distributed Computing in Sensor Networks (DCOSS)*, pp. 165–172, 2006.
- [136] J.-L. Liu and C. V. Ravishankar, “LEACH-GA: Genetic algorithm-based energy-efficient adaptive clustering protocol for wireless sensor networks,” *International Journal of Machine Learning and Computing*, vol. 1, no. 1, pp. 79–85, 2011.

- [137] S. He, Y. Dai, R. Zhou, and S. Zhao, "A Clustering Routing Protocol for Energy Balance of WSN based on Genetic Clustering Algorithm," *IERI Procedia*, vol. 2, pp. 788–793, 2012.
- [138] S. Bayraklı and S. Z. Erdogan, "Genetic algorithm based energy efficient clusters (gabec) in wireless sensor networks," *Procedia Computer Science*, vol. 10, pp. 247–254, 2012.
- [139] S. Hussain and A. W. Matin, "Base Station Assisted Hierarchical Cluster-Based Routing," in *International Conference on Wireless and Mobile Communications (ICWMC'06)*, pp. 1–6, IEEE, 2006.
- [140] A. Peiravi, H. R. Mashhadi, and S. Hamed Javadi, "An optimal energy-efficient clustering method in wireless sensor networks using multi-objective genetic algorithm," *International Journal of Communication Systems*, vol. 26, no. 1, pp. 114–126, 2013.
- [141] H.-S. Seo, S.-J. Oh, and C.-W. Lee, "Evolutionary Genetic Algorithm for Efficient Clustering of Wireless Sensor Networks," in *Proceedings of 6th IEEE Consumer Communications and Networking Conference*, pp. 1–5, 2009.
- [142] S. K. Gupta, P. Kuila, and P. K. Jana, "GAR: an energy efficient GA-based routing for wireless sensor networks," in *International conference on distributed computing and internet technology*, pp. 267–277, Springer, 2013.
- [143] S.-S. Chiang, C.-H. Huang, and K.-C. Chang, "A Minimum Hop Routing Protocol for Home Security Systems Using Wireless Sensor Networks," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 4, pp. 1483–1489, 2007.
- [144] J. R. Srivastava and T. Sudarshan, "A genetic fuzzy system based optimized zone based energy efficient routing protocol for mobile sensor networks (OZEED)," *Applied Soft Computing*, vol. 37, pp. 863–886, 2015.
- [145] J. R. Srivastava and T. Sudarshan, "ZEED: Zone based energy efficient routing protocol for mobile sensor networks," in *IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 990–996, 2013.
- [146] M. Shokouhifar and A. Jalali, "A new evolutionary based application specific routing protocol for clustered wireless sensor networks," *AEU-International Journal of Electronics and Communications*, vol. 69, no. 1, pp. 432–441, 2015.
- [147] S. H. Kang and T. Nguyen, "Distance based thresholds for cluster head selection in wireless sensor networks," *IEEE Communications Letters*, vol. 16, no. 9, pp. 1396–1399, 2012.
- [148] J.-G. Jia, Z.-W. He, J.-M. Kuang, and Y.-h. Mu, "An energy consumption balanced clustering algorithm for wireless sensor network," in *Proceedings of 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM)*, pp. 1–4, IEEE, 2010.

- [149] B. Barekatain, S. Dehghani, and M. Pourzaferani, "An energy-aware routing protocol for wireless sensor networks based on new combination of genetic algorithm & k-means," *Procedia Computer Science*, vol. 72, pp. 552–560, 2015.
- [150] T. Bhatia, S. Kansal, S. Goel, and A. Verma, "A genetic algorithm based distance-aware routing protocol for wireless sensor networks," *Computers & Electrical Engineering*, vol. 56, pp. 441–455, 2016.
- [151] K. Naregal and A. Gudnavar, "Improved cluster routing protocol for wireless sensor network through simplification," in *Proceedings of 18th IEEE International Conference on Advanced Computing and Communications (ADCOM)*, pp. 1–3, 2012.
- [152] S. Bhushan, R. Pal, and S. G. Antoshchuk, "Energy Efficient Clustering Protocol for Heterogeneous Wireless Sensor Network: A Hybrid Approach using GA and K-means," in *IEEE Second International Conference on Data Stream Mining & Processing (DSMP)*, pp. 381–385, 2018.
- [153] G. Smaragdakis, I. Matta, and A. Bestavros, "SEP: A stable election protocol for clustered heterogeneous wireless sensor networks," tech. rep., Boston University Computer Science Department, 2004.
- [154] J. Bhola, S. Soni, and G. K. Cheema, "Genetic algorithm based optimized leach protocol for energy efficient wireless sensor networks," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, pp. 1–8, 2019.
- [155] Y. Cooren, *Perfectionnement d'un algorithme adaptatif d'Optimisation par Essaim Particulaire: application en génie médical et en électronique*. PhD thesis, Université Paris12 VAL DE MARNE, France, November 2008.
- [156] T. Gui, C. Ma, F. Wang, and D. E. Wilkins, "Survey on swarm intelligence based routing protocols for wireless sensor networks: An extensive study," in *IEEE International Conference on Industrial Technology (ICIT)*, pp. 1944–1949, 2016.
- [157] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm intelligence*, vol. 1, no. 1, pp. 33–57, 2007.
- [158] N. A. Latiff, C. C. Tsimenidis, and B. S. Sharif, "Energy-aware clustering for wireless sensor networks using particle swarm optimization," in *Proceedings 18th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1–5, IEEE, 2007.
- [159] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on wireless communications*, vol. 1, no. 4, pp. 660–670, 2002.
- [160] R. S. Elhabyan and M. C. Yagoub, "PSO-HC: Particle swarm optimization protocol for hierarchical clustering in Wireless Sensor Networks," in *Proceedings of 10th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pp. 417–424, IEEE, 2014.

- [161] Y. Ding, Y. Hu, K. Hao, and L. Cheng, "MPSICA: An intelligent routing recovery scheme for heterogeneous wireless sensor networks," *Information Sciences*, vol. 308, pp. 49–60, 2015.
- [162] D. Kumar, T. C. Aseri, and R. Patel, "EEHC: Energy Efficient Heterogeneous Clustered scheme for wireless sensor networks," *Computer Communications*, vol. 32, no. 4, pp. 662 – 667, 2009.
- [163] A. Boukerche, A. Martirosyan, and R. Pazzi, "An inter-cluster communication based energy aware and fault tolerant protocol for wireless sensor networks," *Mobile Networks and Applications*, vol. 13, no. 6, pp. 614–626, 2008.
- [164] R. S. Elhabyan and M. C. Yagoub, "Two-tier particle swarm optimization protocol for clustering and routing in wireless sensor network," *Journal of Network and Computer Applications*, vol. 52, pp. 116–128, 2015.
- [165] S. Tyagi, S. K. Gupta, S. Tanwar, and N. Kumar, "EHE-LEACH: Enhanced heterogeneous LEACH protocol for lifetime enhancement of wireless SNs," in *Proceedings of International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 1485–1490, August 2013.
- [166] A. Rahmanian, H. Omranpour, M. Akbari, and K. Raahemifar, "A novel genetic algorithm in LEACH-C routing protocol for sensor networks," in *Proceedings of 24th Canadian Conference on Electrical and Computer Engineering (CCECE)*, pp. 1096–1100, May 2011.
- [167] P. S. Rao, P. K. Jana, and H. Banka, "A particle swarm optimization based energy efficient cluster head selection algorithm for wireless sensor networks," *Wireless Networks*, vol. 23, no. 7, pp. 2005–2020, 2017.
- [168] A. Bari, A. Jaekel, and S. Bandyopadhyay, "Clustering strategies for improving the lifetime of two-tiered sensor networks," *Computer Communications*, vol. 31, no. 14, pp. 3451–3459, 2008.
- [169] T. W. Tukisi, T. N. Mathaba, and M. O. Odhiambo, "Multi-hop PSO based routing protocol for Wireless Sensor Networks with Energy Harvesting," in *IEEE International Conference on Information Communications Technology and Society (ICTAS)*, pp. 1–6, 2019.
- [170] A. A. Hady, S. M. A. El-Kader, and H. S. Eissa, "Intelligent sleeping mechanism for wireless sensor networks," *Egyptian Informatics Journal*, vol. 14, no. 2, pp. 109–115, 2013.
- [171] A. Coloni, M. Dorigo, and V. Maniezzo, "An Investigation of some Properties of an Ant Algorithm," in *Proceedings of the Parallel Problem Solving From Nature Conference (PPSN'92)*, no. 509–520, 1992.
- [172] S. Selvakennedy, S. Sinnappan, and Y. Shang, "T-ANT: a nature-inspired data gathering protocol for wireless sensor networks," *Journal of Communications*, vol. 1, no. 2, pp. 22–29, 2006.

- [173] S. Selvakennedy and S. Sinnappan, “The time-controlled clustering algorithm for optimized data dissemination in wireless sensor network,” in *Proceedings of IEEE Conference on Local Computer Networks 30th Anniversary (LCN’05)*, pp. 2–pp, IEEE, 2005.
- [174] L. Wang, R. Zhang, and S. Geng, “An energy-balanced ant-based routing protocol for wireless sensor networks,” in *IEEE 5th International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 1–4, IEEE, 2009.
- [175] M. Ziyadi, K. Yasami, and B. Abolhassani, “Adaptive clustering for energy efficient wireless sensor networks based on ant colony optimization,” in *Proceedings of 7th IEEE Annual Communication Networks and Services Research Conference*, pp. 330–334, 2009.
- [176] J. Yang, M. Xu, W. Zhao, and B. Xu, “A multipath routing protocol based on clustering and ant colony optimization for wireless sensor networks,” *Sensors*, vol. 10, no. 5, pp. 4521–4540, 2010.
- [177] D. Ganesan, R. Govindan, S. Shenker, and D. Estrin, “Highly resilient, energy efficient multipath routing in wireless sensor networks,” *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 4, pp. 10–24, 2002.
- [178] Z. Tu, Q. Wang, and Y. Shen, “Optimal mobile agent routing for data fusion in distributed sensor networks using improved ant colony algorithm,” in *IEEE Instrumentation and Measurement Technology Conference*, pp. 155–159, 2008.
- [179] Z. Zou and Y. Qian, “Wireless sensor network routing method based on improved ant colony algorithm,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 3, pp. 991–998, 2019.
- [180] D. Karaboga, “An idea based on honey bee swarm for numerical optimization,” tech. rep., Technical report-tr06, Erciyes University, Engineering Faculty, Computer . . . , 2005.
- [181] D. Karaboga and B. Basturk, “On the performance of artificial bee colony (ABC) algorithm,” *Applied soft computing*, vol. 8, no. 1, pp. 687–697, 2008.
- [182] P. Loubiere, *Amélioration des métaheuristiques d’optimisation à l’aide de l’analyse de sensibilité*. PhD thesis, Université Paris Est, November 2016.
- [183] J. Kumar, S. Tripathi, and R. K. Tiwari, “A survey on routing protocols for wireless sensor networks using swarm intelligence,” *International Journal of Internet Technology and Secured Transactions*, vol. 6, no. 2, pp. 79–102, 2016.
- [184] X. Cai, Y. Duan, Y. He, J. Yang, and C. Li, “Bee-sensor-C: an energy-efficient and scalable multipath routing protocol for wireless sensor networks,” *International Journal of Distributed Sensor Networks*, vol. 2015, pp. Article ID 976127, 14 pages, March 2015.

- [185] D. Karaboga, S. Okdem, and C. Ozturk, "Cluster based wireless sensor network routing using Artificial Bee Colony algorithm," *Wireless Networks*, vol. 18, no. 7, pp. 847–860, 2012.
- [186] M. Saleem and M. Farooq, "Beesensor: a bee-inspired power aware routing protocol for wireless sensor networks," in *Workshops on Applications of Evolutionary Computation*, pp. 81–90, Springer, 2007.
- [187] T. Camilo, C. Carreto, J. S. Silva, and F. Boavida, "An energy-efficient ant-based routing algorithm for wireless sensor networks," in *International Workshop on Ant Colony Optimization and Swarm Intelligence*, pp. 49–59, Springer, 2006.
- [188] Y. Zhang, L. D. Kuhn, and M. P. Fromherz, "Improvements on ant routing for sensor networks," in *International Workshop on Ant Colony Optimization and Swarm Intelligence*, pp. 154–165, Springer, 2004.
- [189] P. S. Mann, S. Singh, and A. Kumar, "Computational intelligence based meta-heuristic for energy-efficient routing in wireless sensor networks," in *IEEE Congress on Evolutionary Computation (CEC)*, pp. 4460–4467, IEEE, 2016.
- [190] A. A. Bara'a and E. A. Khalil, "A new evolutionary based routing protocol for clustered heterogeneous wireless sensor networks," *Applied Soft Computing*, vol. 12, no. 7, pp. 1950–1957, 2012.
- [191] A. A. A. Ari, B. O. Yenke, N. Labraoui, I. Damakoa, and A. Gueroui, "A power efficient cluster-based routing algorithm for wireless sensor networks: Honeybees swarm intelligence based approach," *Journal of Network and Computer Applications*, vol. 69, pp. 77–97, 2016.
- [192] J.-F. Roy, *Apprentissage automatique avec garanties de généralisation à l'aide de méthodes d'ensemble maximisant le désaccord*. PhD thesis, Université Laval, Canada, May 2018.
- [193] M. A. Alsheikh, S. Lin, D. Niyato, and H.-P. Tan, "Machine learning in wireless sensor networks: Algorithms, strategies, and applications," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 4, pp. 1996–2018, 2014.
- [194] I. Gupta, D. Riordan, and S. Sampalli, "Cluster-head election using fuzzy logic for wireless sensor networks," in *3rd IEEE Annual Communication Networks and Services Research Conference (CNSR'05)*, pp. 255–260, 2005.
- [195] J. Kim, S. Park, Y. Han, and T. Chung, "CHEF: Cluster Head Election mechanism using Fuzzy logic in Wireless Sensor Networks," in *Proceedings of 10th International Conference on Advanced Communication Technology*, vol. 1, pp. 654–659, February 2008.
- [196] G. Ran, H. Zhang, and S. Gong, "Improving on LEACH protocol of wireless sensor networks using fuzzy logic," *Journal of Information & Computational Science*, vol. 7, no. 3, pp. 767–775, 2010.

- [197] F. Zhang, Q.-Y. Zhang, and Z.-M. Sun, "ICT2TSK: An improved clustering algorithm for WSN using a type-2 Takagi-Sugeno-Kang Fuzzy Logic System," in *IEEE Symposium on Wireless Technology & Applications (ISWTA)*, pp. 153–158, IEEE, 2013.
- [198] A. Pires, C. Silva, E. Cerqueira, D. Monteiro, and R. Viegas, "CHEATS: A cluster-head election algorithm for WSN using a Takagi-Sugeno fuzzy system," in *IEEE Third Latin-American Conference on Communications*, pp. 1–6, 2011.
- [199] C. Li, M. Ye, G. Chen, and J. Wu, "An energy-efficient unequal clustering mechanism for wireless sensor networks," in *Proceedings of IEEE International Conference on Mobile Adhoc and Sensor Systems Conference*, pp. 8–pp, 2005.
- [200] B. Mostafa, C. Saad, and H. Abderrahmane, "Fuzzy logic approach to improving stable election protocol for clustered heterogeneous wireless sensor networks," *J. Theor. Appl. Inf. Technol.*, vol. 14, pp. 112–116, 2014.
- [201] H. Bagci and A. Yazici, "An energy aware fuzzy approach to unequal clustering in wireless sensor networks," *Applied Soft Computing*, vol. 13, no. 4, pp. 1741–1749, 2013.
- [202] A. Alaybeyoglu, "A distributed fuzzy logic-based root selection algorithm for wireless sensor networks," *Computers & Electrical Engineering*, vol. 41, pp. 216–225, 2015.
- [203] C.-Y. Wen and W. A. Sethares, "Adaptive decentralized re-clustering for wireless sensor networks," in *IEEE International Conference on Systems, Man and Cybernetics*, vol. 4, pp. 2709–2716, 2006.
- [204] Z. M. Zahedi, R. Akbari, M. Shokouhifar, F. Safaei, and A. Jalali, "Swarm intelligence based fuzzy routing protocol for clustered wireless sensor networks," *Expert Systems with Applications*, vol. 55, pp. 313–328, 2016.
- [205] R. Logambigai and A. Kannan, "Fuzzy logic based unequal clustering for wireless sensor networks," *Wireless Networks*, vol. 22, no. 3, pp. 945–957, 2016.
- [206] Y. Zhang, J. Wang, D. Han, H. Wu, and R. Zhou, "Fuzzy-Logic Based Distributed Energy-Efficient Clustering Algorithm for Wireless Sensor Networks," *Sensors*, vol. 17, no. 7, p. Article 1554 (21 pages), 2017.
- [207] M. Liu, Y. Zheng, J. Cao, G. Chen, L. Chen, and H.-g. Gong, "An energy-aware protocol for data gathering applications in wireless sensor networks," in *IEEE International Conference on Communications*, pp. 3629–3635, 2007.
- [208] M. Bhanderi and H. Shah, "Machine learning for wireless sensor network: A review, challenges and applications," *Adv. Electron. Electr. Eng.*, vol. 4, pp. 475–486, 2014.
- [209] B. Solaiman and A. Sheta, "Computational intelligence for wireless sensor networks: Applications and clustering algorithms," *International Journal of Computer Applications*, vol. 73, no. 15, pp. 1–8, 2013.

- [210] G. Wu, H. Li, and L. Yao, "A group-based mobile agent routing protocol for multitype wireless sensor networks," in *Proceedings of IEEE/ACM International Conference on Green Computing and Communications & International Conference on Cyber, Physical and Social Computing*, pp. 42–49, 2010.
- [211] A. Arya, A. Malik, and R. Garg, "Reinforcement learning based routing protocols in WSNs: A survey," *Int. J. Comput. Sci. Eng. Technol*, vol. 4, pp. 1401–1404, 2013.
- [212] A. Forster and A. L. Murphy, "CLIQUE: Role-free clustering with Q-learning for Wireless Sensor Networks," in *Proceedings of 29th IEEE International Conference on Distributed Computing Systems*, pp. 441–449, 2009.
- [213] S. Bandyopadhyay and E. J. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," in *Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003)*, vol. 3, pp. 1713–1723, IEEE, 2003.
- [214] B. Debowski, P. Spachos, and S. Areibi, "Q-learning enhanced gradient based routing for balancing energy consumption in WSNs," in *21st IEEE International Workshop on Computer Aided Modelling and Design of Communication Links and Networks (CAMAD)*, pp. 18–23, 2016.
- [215] W. Guo, C. Yan, and T. Lu, "Optimizing the lifetime of wireless sensor networks via reinforcement-learning-based routing," *International Journal of Distributed Sensor Networks*, vol. 15, no. 2, p. 20 pages, 2019.
- [216] S. Yessad, N. Tazarart, L. Bakli, L. Medjkoune-Bouallouche, and D. Aissani, "Balanced energy efficient routing protocol for wsn," in *International Conference on Communications and Information Technology (ICCIT)*, pp. 326–330, IEEE, 2012.
- [217] J. A. Boyan and M. L. Littman, "Packet routing in dynamically changing networks: A reinforcement learning approach," in *Advances in neural information processing systems*, pp. 671–678, 1994.
- [218] A. P. Renold and S. Chandrakala, "MRL-SCSO: multi-agent reinforcement learning-based self-configuration and self-optimization protocol for unattended wireless sensor networks," *Wireless Personal Communications*, vol. 96, no. 4, pp. 5061–5079, 2017.
- [219] S. O. Haykin, *Neural Networks and Learning Machines*. McMaster University, Canada: Pearson; 3 edition, 2009.
- [220] H. White, "Learning in artificial neural networks: A statistical perspective," *Neural computation*, vol. 1, no. 4, pp. 425–464, 1989.
- [221] M. Cordina and C. J. Debono, "Increasing wireless sensor network lifetime through the application of som neural networks," in *Proceedings of the 3rd IEEE International Symposium on Communications, Control and Signal Processing*, pp. 467–471, 2008.

- [222] N. K. Nehra, M. Kumar, and R. Patel, "Neural network based energy efficient clustering and routing in wireless sensor networks," in *Proceedings of the 1st IEEE International Conference on Networks & Communications*, pp. 34–39, 2009.
- [223] S. Yi, J. Heo, Y. Cho, and J. Hong, "PEACH: Power-efficient and adaptive clustering hierarchy protocol for wireless sensor networks," *Computer communications*, vol. 30, no. 14-15, pp. 2842–2852, 2007.
- [224] A. Mehmood, Z. Lv, J. Lloret, and M. M. Umar, "ELDC: An Artificial Neural Network Based Energy-Efficient and Robust Routing Scheme for Pollution Monitoring in WSNs," *IEEE Transactions on Emerging Topics in Computing*, vol. 8, no. 1, pp. 106–114, 2020.
- [225] A. Mehmood, S. Khan, B. Shams, and J. Lloret, "Energy-efficient multi-level and distance-aware clustering mechanism for WSNs," *International Journal of Communication Systems*, vol. 28, no. 5, pp. 972–989, 2015.
- [226] O. Younis and S. Fahmy, "HEED: a Hybrid, Energy-Efficient, Distributed clustering approach for ad hoc sensor networks," *IEEE Transactions on mobile computing*, vol. 3, no. 4, pp. 366–379, 2004.
- [227] N. Gautam, W.-I. Lee, and J.-Y. Pyun, "Dynamic clustering and distance aware routing protocol for wireless sensor networks," in *Proceedings of the 6th ACM Symposium on Performance Evaluation of Wireless Ad hoc, Sensor, and Ubiquitous Networks*, pp. 9–14, 2009.
- [228] J. P. Ortega, M. Del, R. B. Rojas, and M. J. Somodevilla, "Research issues on K-means algorithm: An experimental trial using Matlab," in *Proceedings CEUR Workshop : Semantic Web and new Technologies*, pp. 83–96, 2009.
- [229] P. Sasikumar and S. Khara, "K-means Clustering in Wireless Sensor Networks," in *Proceedings of 4th IEEE International Conference on Computational Intelligence and Communication Networks*, pp. 140–144, 2012.
- [230] J. Yunjie, L. Ming, Z. Song, and D. Pengtao, "A Clustering Routing Algorithm based on Energy and Distance in WSN," in *Proceedings of IEEE International Conference on Computer Distributed Control and Intelligent Environmental Monitoring*, pp. 9–12, 2012.
- [231] A. Gachhadar and O. N. Acharya, "K-means based Energy Aware Clustering Algorithm in Wireless Sensor Network," *International Journal of Scientific & Engineering Research*, vol. 5, no. 5, pp. 156–161, 2014.
- [232] E. Rabiaa, B. Noura, and C. Adnene, "Improvements in LEACH based on K-means and Gauss algorithms," *Procedia Computer Science*, vol. 73, pp. 460–467, 2015.
- [233] S. Randhawa and S. Jain, "Performance Analysis of LEACH with Machine Learning Algorithms in Wireless Sensor Networks," *International Journal of Computer Applications*, vol. 147, no. 2, pp. 7–12, 2016.

- [234] I. Quchan, “Towards Energy efficient K-means based Clustering Scheme for Wireless Sensor Networks,” *International Journal of Grid and Distributed Computing*, vol. 9, no. 7, pp. 265–276, 2016.
- [235] E. Rezaei, A. A. Baradaran, and A. Heydariyan, “Multi-hop routing algorithm using Steiner points for reducing energy consumption in Wireless Sensor Networks,” *Wireless Personal Communications*, vol. 86, no. 3, pp. 1557–1570, 2016.
- [236] I. M. Butt and S. Khan, “Analyzing & enhancing energy efficient communication protocol for wireless micro-sensor networks,” in *Proceedings of IEEE International Conference on Information and Communication Technologies*, pp. 323–327, 2005.
- [237] M. R. Senouci, A. Mellouk, H. Senouci, and A. Aissani, “Performance evaluation of network lifetime spatial-temporal distribution for WSN routing protocols,” *Journal of Network and Computer Applications*, vol. 35, no. 4, pp. 1317–1328, 2012.
- [238] E. Rezaei, A. A. Baradaran, and A. Heydariyan, “TDTTCGE: Two-dimensional technique based on center of gravity and energy center in wireless sensor network,” *Journal of Basic and Applied Scientific Research (JBASR)*, vol. 3, no. 8, pp. 194–201, 2013.
- [239] A. Mahboub, M. Arioua, and E.-M. En-Naimi, “Energy-efficient hybrid K-means algorithm for clustered Wireless Sensor Networks,” *International Journal of Electrical and Computer Engineering*, vol. 7, no. 4, pp. 2054–2060, 2017.
- [240] M. Abderrahim, H. Hakim, H. Boujemaa, and F. Touati, “A Clustering Routing based on Dijkstra Algorithm for WSNs,” in *Proceedings of 19th IEEE International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, pp. 605–610, 2019.
- [241] H. Zhang and Y. Jiang, “A Genetic Algorithm Application on Wireless Sensor Networks,” in *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pp. 106–117, Springer, 2014.
- [242] J. Qin, F. Li, S. Mou, and Y. Kang, “Multi-timer based event synchronization control for sensor networks and its application,” *IEEE Transactions on Industrial Electronics*, vol. 63, no. 12, pp. 7765–7775, 2016.
- [243] S. Kurt, H. Yildiz, M. Yigit, B. Tavli, and V. Gungor, “Packet size optimization in wireless sensor networks for smart grid applications,” *IEEE Transactions on Industrial Electronics*, vol. 64, no. 3, pp. 2392–2401, 2016.
- [244] J. C. da Silva Carvalho, A. R. da Rocha, F. Sousa, A. B. B. Torres, and J. N. de Souza, “Management of multiple applications for shared sensor and actuators networks,” *IEEE Latin America Transactions*, vol. 14, no. 10, pp. 4358–4366, 2016.
- [245] W. Liu, T. Deng, H. Jiang, X. Liao, J. Liu, B. Li, and G. Jiang, “Towards robust surface skeleton extraction and its applications in 3d wireless sensor networks,” *IEEE/ACM Transactions on Networking*, vol. 24, no. 6, pp. 3300–3313, 2016.

-
- [246] M. Amjad, M. Sharif, M. K. Afzal, and S. W. Kim, "Tinyos-new trends, comparative views, and supported sensing applications: A review," *IEEE Sensors Journal*, vol. 16, no. 9, pp. 2865–2889, 2016.
- [247] B. Rashid and M. Rehmani, "Applications of wireless sensor networks for urban areas: A survey," *Journal of network and computer applications*, vol. 60, pp. 192–219, 2016.
- [248] G. Han, J. Jiang, M. Guizani, and J. J. C. Rodrigues, "Green routing protocols for wireless multimedia sensor networks," *IEEE Wireless Communications*, vol. 23, no. 6, pp. 140–146, 2016.
- [249] J. Yan, M. Zhou, and Z. Ding, "Recent advances in energy-efficient routing protocols for wireless sensor networks: A review," *IEEE Access*, vol. 4, pp. 5673–5686, 2016.
- [250] X. Liu, "A typical hierarchical routing protocols for wireless sensor networks: A review," *IEEE Sensors Journal*, vol. 15, no. 10, pp. 5372–5383, 2015.
- [251] V. Kumar, S. B. Dhok, R. Tripathi, and S. Tiwari, "A review study on analytical estimation of optimal number of clusters in wireless sensor networks," *Transactions on Networks and Communications*, vol. 2, no. 5, pp. 75–103, 2014.
- [252] A. Ng, "Clustering with the k-means algorithm," *Machine Learning*, 2012.
- [253] P. Bholowalia and A. Kumar, "EBK-Means: A clustering technique based on elbow method and k-means in WSN," *International Journal of Computer Applications*, vol. 105, no. 9, pp. 17–24, 2014.
- [254] J. Wang, K. Wang, J. Niu, and W. Liu, "A K-medoids based clustering algorithm for wireless sensor networks," in *Proceedings of IEEE International Workshop on Advanced Image Technology (IWAIT)*, pp. 1–4, IEEE, 2018.
- [255] S. Naeimi, C.-O. Chow, and H. Ishii, "Directional multi-hop clustering routing protocol for wireless sensor networks," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 14, no. 2, pp. 123–134, 2013.
- [256] I. F. Akyildiz and I. H. Kasimoglu, "Wireless sensor and actor networks: research challenges," *Ad hoc Networks*, vol. 2, no. 4, pp. 351–367, 2004.
- [257] O. Boyinbode, H. Le, A. Mbogho, M. Takizawa, and R. Poliah, "A survey on clustering algorithms for wireless sensor networks," in *Proceedings of the 13th IEEE International Conference on Network-based Information Systems*, pp. 358–364, IEEE, 2010.
- [258] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, "Energy-efficient routing protocols in wireless sensor networks: A survey," *IEEE Communications surveys & tutorials*, vol. 15, no. 2, pp. 551–591, 2012.
- [259] M. H. Anisi, A. H. Abdullah, and S. A. Razak, "Energy-efficient and reliable data delivery in wireless sensor networks," *Wireless Networks*, vol. 19, no. 4, pp. 495–505, 2013.

- [260] M. Gavhale and P. D. Saraf, "Survey on algorithms for efficient cluster formation and cluster head selection in MANET," *Procedia Computer Science*, vol. 78, no. C, pp. 477–482, 2016.
- [261] W.-S. Jung, K.-W. Lim, Y.-B. Ko, and S.-J. Park, "Efficient clustering-based data aggregation techniques for wireless sensor networks," *Wireless Networks*, vol. 17, no. 5, pp. 1387–1400, 2011.
- [262] H. Lakhlef, "A multi-level clustering scheme based on cliques and clusters for wireless sensor networks," *Computers & Electrical Engineering*, vol. 48, pp. 436–450, 2015.
- [263] S. M. Bozorgi, A. S. Rostami, A. A. R. Hosseinabadi, and V. E. Balas, "A new clustering protocol for energy harvesting-wireless sensor networks," *Computers & Electrical Engineering*, vol. 64, pp. 233–247, 2017.
- [264] C. Sivaraaj, P. J. A. Alphonse, and T. N. Janakiraman, "Energy-efficient and load distributed clustering algorithm for dense wireless sensor networks," *International Journal of Intelligent Systems and Applications*, vol. 9, no. 5, p. 34, 2017.
- [265] M. Busse, T. Haenselmann, and W. Effelsberg, "TECA: atology and energy control algorithm for wireless sensor networks," in *Proceedings of the 9th ACM International Symposium on Modeling and Simulation of Wireless and Mobile Systems*, pp. 317–321, ACM, 2006.
- [266] Y. Chen and S. H. Son, "A fault-tolerant topology control in wireless sensor networks," in *Proceedings of the 3rd ACS/IEEE International Conference on Computer Systems and Applications, 2005*, p. 57, IEEE, 2005.
- [267] R. Ghosh and S. Basagni, "Nappingbackbones: energy efficient topology control for wireless sensor networks," in *Proceedings of IEEE Radio and Wireless Symposium*, pp. 611–614, IEEE, 2006.
- [268] M. B. Benmahdi and M. Lehsaini, "A GA-based Multihop Routing Scheme using K-Means Clustering approach for Wireless Sensor Networks," in *The Second International Conference on Embedded & Distributed Systems, EDiS'2020*, p. Accepted paper, April 2020.
- [269] N. R. Roy and P. Chandra, "EEDAC-WSN: Energy Efficient Data Aggregation in Clustered WSN," in *Proceedings of IEEE International Conference on Automation, Computational and Technology Management (ICACTM)*, pp. 586–592, IEEE, 2019.
- [270] A. Mahboub, E. M. En-Naimi, M. Arioua, and H. Anas, "Performance Evaluation of Cluster Validity Methods an Energy Optimization in Wireless Sensor Networks Using Hybrid K-Medoids Algorithm," in *Proceedings of the 2nd International Conference on Big Data, Cloud and Applications*, pp. 1–7, ACM, 2017.
- [271] C. P. Gupta and A. Kumar, "Optimal number of cluster in wireless sensor network with mobile sink," *International Journal of Scientific and Engineering Research*, vol. 4, no. 8, pp. 1706–1710, 2013.

- [272] M. Abo-Zahhad, S. M. Ahmed, N. Sabor, and S. Sasaki, “A new energy-efficient adaptive clustering protocol based on genetic algorithm for improving the lifetime and the stable period of wireless sensor networks,” *International Journal of Energy, Information and Communications*, vol. 5, no. 3, pp. 47–72, 2014.
- [273] G.-S. Yao, Z.-X. Dong, W.-M. Wen, and Q. Ren, “A Routing Optimization Strategy for Wireless Sensor Networks Based on Improved Genetic Algorithm,” *Journal of Applied Science and Engineering*, vol. 19, no. 2, pp. 221–228, 2016.
- [274] S. Babaie and A. Khadem, “A new method for improving life time in wireless sensor network by using genetic algorithm,” in *Instrumentation, Measurement, Circuits and Systems*, pp. 177–184, Springer, 2012.
- [275] A. Abro, D. Zhongliang, K. A. Memon, and N. ul Ain, “Novel Genetic Algorithm with Efficient Routing Paradigm for Multi-hop WSNs,” in *Proceedings of the 9th IEEE International Conference on Electronics Information and Emergency Communication (ICEIEC)*, pp. 28–31, IEEE, 2019.
- [276] V. K. Singh and V. Sharma, “Elitist genetic algorithm based energy efficient routing scheme for wireless sensor networks,” *International Journal Of Advanced Smart Sensor Network Systems (IJASSN)*, vol. 2, no. 2, 2012.

Résumé

La réduction de la consommation d'énergie et le passage à l'échelle sont des exigences clés dans les réseaux de capteurs sans fil (RCSFs), car ces réseaux sont généralement composés d'un grand nombre de capteurs sous la contrainte énergétique. Par conséquent, l'efficacité énergétique dans ce type de réseaux est considérée comme un problème critique. Une façon d'atteindre cet objectif est de minimiser le nombre de données redondantes envoyées à la station de base à travers l'approche de clustering qui est l'une des meilleures approches en termes d'efficacité énergétique dans les RCSFs à grande échelle. Dans cette thèse, nous avons proposé des solutions économes en énergie pour les RCSFs à grande échelle. Ces solutions se basent sur une amélioration de l'approche de l'apprentissage non supervisé (K-Means) et impliquent des méthodes pour déterminer le nombre de clusters approprié (Silhouette, Elbow et "Rule of Thumb"). Dans la première contribution, nous avons évalué chacune de ces méthodes dans le but de connaître l'approche la plus adéquate pour déterminer le nombre de clusters. Dans la deuxième contribution, nous avons proposé un schéma de routage basé sur une version améliorée de K-Means. La troisième contribution est un schéma de routage basé sur un clustering dynamique et la quatrième contribution est un schéma de routage qui implique "Rule of Thumb" pour déterminer le nombre de CHs, K-Means pour organiser le réseau en clusters et un algorithme génétique amélioré pour établir les chemins entre chaque CH et la station de base.

Les schémas de routage proposés ont été développés sous Matlab. Les résultats de simulations ont montré les avantages de nos solutions en termes de consommation d'énergie, de durée de vie et de passage à l'échelle comparées à d'autres schémas de routage.

Mots clés : Silhouette, Elbow, K-Means, Rule of Thumb, Clustering, RCSFs, Algorithme génétique, Routage à grande échelle.

Abstract

Reducing energy consumption and scalability are key requirements in wireless sensor networks (WSNs), as these networks are generally composed of a large number of sensors under energy constraint. Therefore, energy efficiency in this type of networks is considered a critical problem. One way to achieve this goal is to minimize the amount of redundant data sent to the base station through the clustering approach which is one of the best approaches in terms of energy efficiency in large-scale RCSFs. In this thesis, we have proposed energy-efficient solutions for large-scale RCSFs. These solutions are based on an improvement of the unsupervised learning approach (K-Means) and imply methods to determine the appropriate number of clusters (Silhouette, Elbow and "Rule of Thumb"). In the first contribution, we evaluated each of these methods in order to know the most suitable approach for determining the number of clusters. In the second contribution, we proposed a routing scheme based on an improved version of K-Means. The third contribution is a routing scheme based on dynamic clustering and the fourth contribution is a routing scheme which involves "Rule of Thumb" to determine the number of CHs, K-Means to organize the network into clusters and an improved genetic algorithm to establish the paths between each CH and the base station.

The proposed routing schemes were developed over Matlab. Simulation results have shown the benefits of our solutions in terms of power consumption, lifetime and scalability compared to other routing schemes.

Keywords: Silhouette, Elbow, K-Means, Rule of Thumb, Clustering, WSNs, Genetic algorithm, Large-scale routing.

ملخص

يعد تقليل استهلاك الطاقة وتوسيع النطاق متطلبات رئيسية في شبكات الاستشعار اللاسلكية (WSNs)، حيث تتكون هذه الشبكات بشكل عام من عدد كبير من أجهزة الاستشعار تحت قيود الطاقة. لذلك، يعتبر ترشيد استهلاك الطاقة في هذا النوع من الشبكات مشكلة حرجية. تتمثل إحدى طرق تحقيق هذا الهدف في تقليل كمية المعطيات المتكررة المرسل إلى المحطة الأساسية من خلال نهج التجميع الذي يعد واحدًا من أفضل الأساليب من حيث ترشيد استهلاك الطاقة في شبكات الاستشعار اللاسلكية المنتشرة على أوسع نطاق. في هذه الأطروحة، اقترحنا حلولاً موفرة للطاقة لهذا النوع من الشبكات. تعتمد هذه الحلول على نسخة محسنة من التعلم غير الخاضع للرقابة (K-Means) وإستعمال طرق لتحديد العدد المناسب من المجموعات (Silhouette, Elbow, "Rule of Thumb"). في المساهمة الأولى، قمنا بتقييم كل من هذه الأساليب من أجل معرفة الطريقة الأكثر ملاءمة لتحديد عدد المجموعات. في المساهمة الثانية، اقترحنا مخطط توجيه يعتمد على نسخة محسنة من K-Means. المساهمة الثالثة هي مخطط توجيه يعتمد على التجميع الديناميكي والمساهمة الرابعة عبارة عن مخطط توجيه يتضمن "Rule of Thumb" لتحديد عدد CHs و K-Means لتنظيم الشبكة في مجموعات وخوارزمية وراثية محسنة لتحديد المسارات بين كل CH والمحطة المركزية.

تم تطوير مخططات التوجيه المقترحة باستخدام محاكي Matlab. أظهرت نتائج المحاكاة فوائد حلولنا المقترحة من حيث استهلاك الطاقة وعمر هذه الشبكات وإنتشارها على نطاق أوسع مقارنة بخطة التوجيه الأخرى.

الكلمات المفتاحية: Silhouette, Elbow, K-Means, Rule of Thumb, Clustering, WSNs, خوارزمية جينية، توجيه واسع النطاق.