

N° d'ordre: .....

RÉPUBLIQUE ALGERIENNE DÉMOCRATIQUE ET POPULAIRE  
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ DJILLALI LIABÈS DE SIDI BEL ABBÈS  
FACULTÉ DES SCIENCES EXACTES  
DÉPARTEMENT D'INFORMATIQUE  
LABORATOIRE EEDIS

# THÈSE DE DOCTORAT EN SCIENCES

Filière : Informatique  
Spécialité : Intelligence Artificielle

Par

M<sup>r</sup> MILOUD MIHOUBI

## UNE MÉTAHEURISTIQUE HYBRIDE POUR LE PROBLÈME D'OPTIMISATION DANS LES RÉSEAUX DE CAPTEURS SANS FILS

Soutenue le 10 / 05 / 2021 devant le jury :

Pr. BOUKLI SOFIANE HACÈNE	UDL SBA	Président du jury
Pr. RAHMOUN ABDELLATIF	ESI, SBA	Directeur de thèse
Pr. BENSLIMANE SIDI MOHAMED	ESI, SBA	Examineur
Pr. ADJOUJ RÉDA	UDL SBA	Examineur
Pr. HAMOU MOHAMED RÉDA	Université de Moulay Tahar Saida	Examineur
Pr. LORENZ PASCAL	Université de Haute Alsace, France	Invité

Année Universitaire : 2020 - 2021

*je remercie Allah m'a guidé vers la bonne destination, m'avoir donné l'environnement pour étudier, les possibilités à encaisser et un potentiel pour réussir. Mes sincères salutations sont adressées à mes chers parents qui grâce à eux notre chemin a connu la lumière, des sacrifices et des bénédictions constantes qui m'ont gardé motivé et engagé..*

# REMERCIEMENTS

C E travail ça n'aurait pas été possible d'achever et de rédiger cette thèse sans l'aide, le soutien, la motivation et l'encouragement de mon encadrant le **Professeur : Abdellatif Rahmoun**, professeur de l'école supérieure de l'informatique de Sidi Bel Abbès (Algérie). Je voudrais exprimer ma profonde gratitude à mon directeur de thèse. J'ai appris beaucoup de choses depuis que je suis sous sa supervision. Les bons conseils, le soutien et la patience ont été inestimables tant au niveau académique que personnel, ce dont je lui suis extrêmement reconnaissant. Les connaissances et l'engagement de mon encadrant envers les normes les plus élevées m'ont inspiré et motivé. Un grand merci très spécial adressé au monsieur le **professeur : Lorenz Pascal**, professeur à l'institut universitaire de technologie de Colmar pour son co-Encadrement, son chaleureux accueil dans son groupe de recherche, son suivi infatigable, ses précieux conseils, son aide et son soutien sans fin durant mes années de recherches doctorales. J'apprécie et remercie avec reconnaissance pour le soutien, la motivation, le fructueux débat, la conversation importante et de la bonne orientation de **Monsieur : Nouredine Lasla**, chercheur postdoctoral de Université Hamad bin Khalifa. Je remercie vivement **Monsieur Sofiane BOUKLI HACENE**, Professeur à l'université de Djillali Liabes, Sidi Bel Abbès d'avoir accepté de présider ce Jury. Mes remerciements les plus sincères vont s'adresser aussi à **Monsieur Sidi Mohammed BENSLIMANE**, Professeur à l'école supérieure en informatique, sidi bel abbès. **Monsieur Reda Mohamed Hamou**, Professeur à université Taher Moulay, Saida, et **Monsieur Réda ADJOUJ**, Professeur à l'université de Djillali Liabes, Sidi Bel Abbès, pour avoir accepté d'examiner cette thèse et d'avoir consacré une partie de son temps à la lire à eux tous. Je ne saurais assez les remercier.

## CONTRIBUTION DE L'AUTEUR

1. **M. Miloud**, R. Abdellatif, Z. Meriem, P. Lorenz, B. Lotfi, (2020) : " Intelligent Technique Based On Enhanced Metaheuristic For Optimization Problem In Internet Of Things And Wireless Sensor Network" , International Journal of Grid and High Performance Computing (IJGHPC) , 12(3) , Special Issue Submission : Innovation in Information Technology Through Smart Cloud and Analytics.
2. **M. Miloud**, R. Abdellatif, P. Lorenz. (2019) : "Moth Flame Optimization Algorithm Range-Based for Node Localization Challenge in Decentralized Wireless Sensor Network". International Journal of Distributed Systems and Technologies (IJDST), 10(1), 82-109.doi :10.4018/IJDST.2019010106.
3. **M. Miloud**, R. Abdellatif, P. Lorenz, N. Lasla, (2017) : An effective Bat algorithm for node localization in distributed wireless sensor network . Security and Privacy , 2017 , e20. <https://doi.org/10.1001/spy2.7>.
4. - **M. Mihoubi**, A. Rahmoun, P. Lorenz, (2020) : "Iterative Localization in Decentralized Environment of WSN and IoT" , IEEE International Conference on Communications 7-11 June 2020 , Dublin, Ireland.
5. - **M. Mihoubi**, A. Rahmoun, P. Lorenz, (2019) : "An Enhanced Bat Algorithm for Parallel Localization Based on a Mobile Beacon Sensor in Wireless Sensor Networks", The International Conference on Applied Soft Computing and Communication Networks (ACN'19). Kerala, India during December 18-21, 2019.
6. - B. Lotfi, R. Abdellatif, P. Lorenz, **M. Miloud**, (2019). "Whale Optimization Approach for Optimization Problem In Distributed Wireless Sensor Network", 9th International Conference on Information Systems and Technologies ,Article No. :19 ,pp : 1-7 . <https://doi.org/10.1145/3361570.3361594>.
7. - **M. Mihoubi** , A. Rahmoun, P. Lorenz,(2017) : "Metaheuristic RSSI based for node localization in distributed wireless sensor network". In Global Information Infrastructure and Networking Symposium (GIIS),October25- 27 , 64-70,doi :10.1109/GIIS.2017.8169811.
8. - **M. Miloud** , R. Abdellatif, P. Lorenz (2019) : A Novel Bat Algorithm for Line of Sight Localization In Internet of Things and Wireless Sensor Network , Chapter . 9 : Machine Learning and Cognitive Science Applications in Cyber Security , (301118-033546) .
9. - **M. Mihoubi**, A. Rahmoun, P. Lorenz,. (2020) An Enhanced Bat Algorithm for Parallel Localization Based on a Mobile Beacon Sensor in Wireless Sensor Networks. In : M. Thampi S. et al. (eds) Applied Soft Computing and Communication Networks. ACN 2019. Lecture Notes in Networks and Systems, vol 125. Springer, Singapore.

# TABLE DES MATIÈRES

TABLE DES MATIÈRES	v
LISTE DES FIGURES	viii
LISTE DES TABLEAUX	x
PRÉFACE	1
1 LES ALGORITHMES D'OPTIMISATION MÉTAHEURISTIQUES	4
1.1 INTRODUCTION	4
1.2 PROBLÈME D'OPTIMISATION	5
1.2.1 Formulation du problème d'optimisation	5
1.2.2 Classification des problèmes d'optimisation	5
1.3 LES MÉTAHEURISTIQUES D'OPTIMISATION	6
1.3.1 Concepts des techniques métaheuristiques	7
1.3.2 Classification des techniques métaheuristiques	8
1.4 LES CARACTÉRISTIQUES DES MÉTAHEURISTIQUES :	10
1.5 L'INSPIRATION DES MÉTAHEURISTIQUES :	11
1.5.1 Algorithmes évolutifs :	11
1.5.2 Algorithmes basés sur Swarm Intelligence :	14
1.5.3 Algorithmes d'optimisation basés sur la physique	17
1.6 MOTIVATIONS DE L'HYBRIDATION DES MÉTAHEURISTIQUES	20
1.6.1 Hybridation à usage unique :	20
1.6.2 Hybridation à usages multiples :	20
1.7 TAXONOMIE DES ALGORITHMES HYBRIDES :	20
1.7.1 Hybrides collaboratifs :	20
1.7.2 Hybrides intégratifs :	21
1.8 LES CHALLENGES DES ALGORITHMES HYBRIDES :	22
1.8.1 Convention de dénomination	23
1.8.2 Complexité de l'algorithme hybride :	23
1.8.3 Vitesse de calcul :	23
1.9 CONCLUSION	24
2 LES RÉSEAUX DE CAPTEURS SANS FILS (RCSFs)	25
2.1 INTRODUCTION	27
2.2 HISTORIQUE DES RÉSEAUX DE CAPTEURS SANS FILS	28
2.3 RÉSEAU DE CAPTEURS SANS FIL	30
2.4 PRÉSENTATION D'UN CAPTEUR SANS FIL :	31
2.4.1 Nœud de capteurs sans fil :	31
2.4.2 Type du nœud de capteurs :	32
2.5 DOMAINE D'APPLICATION :	34

2.5.1	Applications militaires . . . . .	35
2.5.2	Applications médicales . . . . .	36
2.5.3	Applications en cyber-sécurité . . . . .	38
2.5.4	Applications environnementales . . . . .	39
2.5.5	Surveillance d'agriculture . . . . .	40
2.5.6	Applications aux réseaux véhiculaires . . . . .	41
2.5.7	Applications aux maisons intelligentes . . . . .	42
2.6	LES EXIGENCES DU RCSF . . . . .	43
2.6.1	L'efficacité énergétique . . . . .	43
2.6.2	L'évolution . . . . .	44
2.6.3	La réactivité . . . . .	44
2.6.4	La fiabilité . . . . .	44
2.6.5	La mobilité . . . . .	44
2.7	LES CARACTÉRISTIQUES DU RCSF . . . . .	45
2.7.1	Coût de production . . . . .	45
2.7.2	Absence d'infrastructure . . . . .	45
2.7.3	Evolution . . . . .	45
2.7.4	Interférences . . . . .	45
2.7.5	Topologie du réseau . . . . .	46
2.7.6	La sécurité physique limitée . . . . .	46
2.7.7	la bande passante limitée . . . . .	46
2.7.8	Le stockage et la récupération . . . . .	46
2.7.9	La durée de vie . . . . .	46
2.8	LES PROBLÈMES D'OPTIMISATION DANS LES RCSFs . . . . .	46
2.8.1	Conception et déploiement . . . . .	47
2.8.2	Agrégation de données . . . . .	48
2.8.3	Protocole de routage . . . . .	49
2.8.4	Clustering . . . . .	50
2.8.5	Qualité de service . . . . .	52
2.8.6	Localisation . . . . .	52
2.9	CONCLUSION . . . . .	54
3	LA LOCALISATION DES NŒUDS DANS LES RCSFs . . . . .	55
3.1	INTRODUCTION . . . . .	55
3.2	PROBLÈME DE LA LOCALISATION : . . . . .	57
3.2.1	Contexte général . . . . .	57
3.2.2	Formulation du problème . . . . .	58
3.3	OBJECTIF DE LOCALISATION . . . . .	59
3.4	CONCEPTION ET ÉVALUATION DE L'ALGORITHME DE LOCALISATION . . . . .	59
3.4.1	La densité des nœuds . . . . .	60
3.4.2	Les facteurs environnementaux . . . . .	60
3.4.3	La topologie du réseau . . . . .	60
3.4.4	Les contraintes de ressources . . . . .	61
3.5	TAXONOMIE DE LOCALISATION DES NŒUDS . . . . .	61
3.5.1	Algorithme à saut unique ou à saut multiples . . . . .	62
3.5.2	Algorithme sans ou basé sur la plage . . . . .	62
3.5.3	Algorithme sans ou basé sur ancre . . . . .	63
3.5.4	Algorithme centralisé ou distribué . . . . .	63
3.6	SYSTÈME DE LOCALISATION . . . . .	63

3.6.1	Estimation distance / angle . . . . .	64
3.6.2	Calcul de position . . . . .	70
3.6.3	Algorithme de localisation . . . . .	72
3.7	ALGORITHME DE LOCALISATION (ÉTAT DE L'ART) . . . . .	72
3.7.1	Les approches mathématiques . . . . .	74
3.7.2	Les approches métaheuristiques . . . . .	75
3.8	CONCLUSION . . . . .	81
<b>4</b>	<b>ALGORITHME D'OPTIMISATION DE LA FLAMME DES MITES</b>	<b>82</b>
4.1	INTRODUCTION . . . . .	82
4.2	ALGORITHME D'OPTIMISATION DE LA FLAMME DES MITES . . . . .	83
4.2.1	Inspiration . . . . .	83
4.2.2	Formulation mathématique . . . . .	84
4.2.3	Génération de la population initiale de papillons . . . . .	87
4.2.4	Mise à jour des positions des papillons . . . . .	88
4.2.5	Mise à jour du nombre de flammes . . . . .	90
4.3	CONCLUSION . . . . .	91
<b>5</b>	<b>LOCALISATION DES NOEUDS À BASE DE L'ALGORITHME DE LA FLAMME DES MITES</b>	<b>92</b>
5.1	INTRODUCTION . . . . .	92
5.2	FORMULATION DU PROBLÈME DE LOCALISATION . . . . .	93
5.3	ÉVALUATION DES PERFORMANCES . . . . .	95
5.3.1	La métrique d'évaluation . . . . .	95
5.3.2	Configuration de la simulation . . . . .	96
5.3.3	Résultats expérimentaux et analyse des performances . . . . .	100
5.3.4	L'effet de la densité des noeuds d'ancrages . . . . .	100
5.3.5	L'effet de la portée de communication . . . . .	102
5.3.6	L'effet de la taille de population . . . . .	108
5.3.7	Etude comparative . . . . .	110
5.3.8	Résultat de localisation sous la portée de communication . . . . .	118
5.3.9	Résultats de localisation sous le nombre d'itération . . . . .	121
5.4	CONCLUSION . . . . .	123
	<b>MES CONTRIBUTIONS SCIENTIFIQUES</b>	<b>124</b>
	<b>CONCLUSION GÉNÉRALE</b>	<b>126</b>
	<b>BIBLIOGRAPHIE</b>	<b>130</b>

# LISTE DES FIGURES

1.1	La phase de croisement . . . . .	13
1.2	La phase de mutation . . . . .	14
1.3	Représentation schématique du mouvement d'une particule dans PSO se déplaçant vers le meilleur global $g^*$ et le meilleur courant $x_i^*$ pour chaque particule $i$ . . . . .	15
1.4	Le cadre collaboratif d'un algorithme hybride, représentant des structures à plusieurs étages, séquentielles et parallèles . . . . .	21
1.5	Structure intégrative d'un algorithme hybride, avec manipulations complètes et partielles. . . . .	22
2.1	Réseau de capteurs sans fil. . . . .	30
2.2	Les composants d'un capteur sans fil. . . . .	31
2.3	Types des capteurs sans fil. . . . .	33
2.4	Suivi de cible sur le champ de bataille. . . . .	36
2.5	Détection d'intrus à la frontière de la région. . . . .	37
2.6	Réseau corporel sans fil. . . . .	38
2.7	Système cyber-physique compatible au RCSF. . . . .	39
2.8	Système de détection automatique d'incendie. . . . .	40
2.9	Réseau de capteur agricole. . . . .	41
2.10	Réseau de capteurs sans fil du réseau routier. . . . .	42
2.11	Réseau de capteurs sans fil d'une maison intelligente. . . . .	43
2.12	Agrégation de données dans des RCSFs. . . . .	48
2.13	Protocole de routage dans le RCSF. . . . .	50
2.14	Clustering dans le RCSF. . . . .	51
2.15	La localisation dans les RCSFs. . . . .	53
3.1	Taxonomie des schémas de localisation pour les réseaux de capteurs. . . . .	61
3.2	Algorithme à saut multiple. . . . .	62
3.3	Schéma de localisation. . . . .	64
3.4	Estimation de la distance à l'aide de ToA. . . . .	67
3.5	Estimation de la distance de ToA basée sur RTT. . . . .	67
3.6	Estimation de distance basée sur TDOA. . . . .	68
3.7	Tri-lateration et Multilateration. . . . .	71
3.8	Triangulation. . . . .	72
3.9	Processus de localisation dans RCSF. . . . .	73
3.10	Phénomène d'ambiguïté. . . . .	77
4.1	Orientation transversale[Mirjalili, 2015c]. . . . .	84
4.2	Trajectoire en spirale autour de sources lumineuses . . . . .	85
4.3	Organigramme de l'algorithme MFOA. . . . .	86

4.4	Tri-lateration et Multilateration. . . . .	91
5.1	Organigramme de la localisation des nœuds. . . . .	94
5.2	Déploiement des nœuds dans le réseau. . . . .	98
5.3	Connectivité de communication entre les nœuds. . . . .	99
5.4	Localisation des nœuds par l'algorithme MFOA. . . . .	99
5.5	L'effet du rapport d'ancrage sur l'erreur de localisation moyenne. . . . .	101
5.6	L'effet du rapport d'ancrage sur le temps de localisation moyenne. . . . .	102
5.7	L'effet de la portée de communication sur l'erreur de locali- sation moyenne. . . . .	103
5.8	Connectivité de communication entre les nœuds de cap- teurs (10 mètres). . . . .	104
5.9	Connectivité de communication entre les nœuds de cap- teurs (20). . . . .	105
5.10	Connectivité de communication entre les nœuds de cap- teurs (30). . . . .	106
5.11	Connectivité de communication entre les nœuds de cap- teurs (40). . . . .	107
5.12	L'effet de la portée de communication sur le temps de loca- lisation moyenne. . . . .	108
5.13	Erreur de localisation par rapport à la taille de la population.	109
5.14	Temps de localisation par rapport à la taille de la population.	110
5.15	Déploiement des nœuds dans le réseau. . . . .	111
5.16	La connectivité de communication entre les nœuds. . . . .	112
5.17	Localisation des nœuds par l'algorithme de PSO . . . . .	113
5.18	Localisation des nœuds par l'algorithme DE . . . . .	114
5.19	Localisation des nœuds par l'algorithme de chauve-souris .	115
5.20	Localisation des nœuds par l'algorithme de GWO . . . . .	116
5.21	Localisation des nœuds par l'algorithme de FPA . . . . .	117
5.22	Localisation des nœuds par l'algorithme de MFOA . . . . .	118
5.23	Erreur de localisation vs plage de communication . . . . .	119
5.24	Temps de localisation vs plage de communication . . . . .	120
5.25	Erreur de localisation vs nombre d'itérations . . . . .	121
5.26	Temps de localisation vs nombre d'itérations . . . . .	122

# LISTE DES TABLEAUX

1.1	Inspiration des algorithmes métaheuristiques . . . . .	19
3.1	Comparaison qualitative des techniques de localisation basées sur l'aire de répartition . . . . .	70
5.1	Configuration des paramètres du RCSF . . . . .	97

# INTRODUCTION GÉNÉRALE

LES réseaux de Capteurs sans Fil (RCSFs) sont des réseaux ad-hoc avec un grand nombre de nœuds qui sont des micros capteurs capables de collecter et de transmettre des données environnementales de manière autonome. Récemment, les réseaux de capteurs sans fil sont devenus un domaine de recherche très actif et sont utilisés dans différents environnements pour obtenir diverses tâches telles que la recherche, les secours en cas de catastrophe, la prédiction climatique, la pression atmosphérique, le suivi des cibles est également un certain nombre de tâches dans les environnements intelligents [Adnan et al., 2014]. Les avancées significatives dans les systèmes micro-électromécanique (MEMS), la technologie de la communication et l'informatique ont motivé le lancement d'un grand nombre d'applications croissantes de réseaux de capteurs sans fil distribués composés de centaines ou de milliers de nœuds. Chaque nœud est capable de détecter l'environnement, de calculer et de communiquer avec une unité centrale ou avec d'autres capteurs dans le système. La topologie du réseau est constituée de l'ensemble des nœuds répartis et distribués de manière aléatoire dans une région de captage, ce qui est un moyen courant de déployer des réseaux de capteurs sans fil. En outre, plusieurs problèmes sont abordés dans le RCSFs, tels que la minimisation de l'énergie, la compression des images, les algorithmes de réseaux auto-organisés, les protocoles de routage, la sécurité, la gestion de la qualité de service [Mao et al., 2007b] et la localisation des nœuds. Le réseau est appelé réseau ad-hoc car il n'y a pas de protocole de communication prioritaire établi [Widmer et al., 2001]. L'établissement d'un algorithme de localisation efficace de conception dépend d'un schéma de localisation réussi afin de calculer la position correcte dans un système de coordonnées. Par exemple, dans un nœud de maternelle, la localisation peut être utilisée pour surveiller l'interaction des enfants avec les jouets, dans le même sens, la localisation peut être appliquée pour le suivi et surveillance des patients à distance. La localisation des nœuds est nécessaire pour signaler dans différentes situations telles que l'origine des événements, aider les requêtes du groupe des capteurs et également pour répondre aux questions sur la couverture du réseau. Par conséquent, la localisation des nœuds est devenue l'un des défis fondamentaux des réseaux de capteurs sans fil [Phoemphon et al., 2018]. Elle joue un rôle essentiel dans plusieurs domaines, tels que la couverture, le déploiement, les informations de routage, le service de localisation, le suivi des cibles et le lancement de mortiers. L'objectif principal de la localisation des nœuds est d'estimer l'emplacement des capteurs avec des informations de localisation initialement inconnues [Patil et al., 2005], pour y parvenir, le processus utilise la connaissance des positions absolues de quelques capteurs et des mesures de l'intercapteur, telles que les

mesures de distance. Dans les réseaux de capteurs sans fil, nous distinguons deux types de nœuds qui sont des nœuds de capteurs d'ancrage et des nœuds de capteurs inconnus. Les nœuds de capteurs dont on connaît les positions sont appelées nœuds ancres, " balises " ou " références ". Aussi, les nœuds d'ancrage contiennent des informations précises sur leur position. En revanche, les nœuds de capteurs inconnus n'ont pas ces spécifications. La capacité d'auto-localisation est hautement souhaitable dans les applications de surveillance de l'environnement telles que la détection d'intrusion, la surveillance du trafic routier, la surveillance de la santé, etc. Une solution intuitive de la localisation des nœuds consiste à équiper chaque nœud avec des dispositifs du système de positionnement global (GPS), mais malheureusement, lorsque le nombre de nœuds de capteurs augmente, l'ajout d'une connexion GPS à tous les nœuds du réseau n'est pas une méthode entièrement réalisable car :

- 1- La méthode GPS ne peut pas être mise en œuvre dans une forêt dense, un terrain montagneux ou d'autres endroits avec une ligne de vue bloquée avec les communications GPS.
- 2- La consommation d'énergie du système GPS diminue la durée de vie de la batterie des nœuds de capteurs, ce qui entraîne une réduction de la durée de vie de l'ensemble du réseau.
- 3- Le coût de production d'un réseau GPS est un problème important avec un grand nombre de nœuds.
- 4- La taille de la couverture GPS et de l'antenne entraîne une augmentation significative de la taille des nœuds de capteurs par un facteur considérable, cependant, la zone / taille des nœuds de capteurs doivent être petits [J et al., 2000].
- 5- La faible précision du service dans des environnements spéciaux (intérieurs, souterrains, etc.).

Les méthodes de localisation ont généralement des modèles en trois phases, comme résumé ci-dessous :

- 1- Calcul de la distance entre les nœuds d'ancrage et les nœuds de capteurs inconnus.
- 2- Calcul de la position des nœuds en se basant sur la distance calculée.
- 3- Détermination de l'emplacement du nœud en utilisant les informations de la distance ou de la distance des nœuds voisins.

Pour la première phase, en premier lieu chaque nœud utilise d'abord la capacité de communication pour réaliser certaines mesures comme l'heure d'arrivée à leurs voisins, estimer la distance à un seul bond après cela en utilisant des méthodes comme les algorithmes de distance distribuée sur le chemin le plus court pour estimer la distance à sauts multiples des nœuds de balise [N et K, 2003]. Pour la deuxième partie, chaque nœud utilise des méthodes telles que la triangulation pour calculer sa position en utilisant des distances de trois nœuds de balises ou plus et ceci basé sur la distance estimée. Pour la dernière partie, chaque nœud capteur trouve la position en fonction des contraintes sur les distances à ses voisins. En fait, le problème de localisation des nœuds a été considéré comme un problème d'optimisation multidimensionnel, où des algorithmes d'optimisation sont utilisés pour résoudre ce problème. Une énorme recherche se basant sur les algorithmes métaheuristiques a été adressé pour optimiser les performances de localisation des nœuds. Dans cette thèse, L'algorithme

d'optimisation appelé " Algorithme d'optimisation de la flamme des mites  
" est proposé pour résoudre le problème de localisation.

# LES ALGORITHMES D'OPTIMISATION MÉTAHEURISTIQUES

1

## 1.1 INTRODUCTION

L'optimisation est la procédure permettant de trouver la meilleure solution à un problème donné parmi toutes les solutions possibles [Khalilpourazari et Khalilpourazary, 2016]. Les plupart des méthodologies d'optimisation traditionnelles sont inefficaces pour résoudre les problèmes complexes d'aujourd'hui. les problèmes d'optimisation difficiles sont définis comme des problèmes qui ne peuvent pas être résolus jusqu'à l'optimalité, ou à toute limite garantie, par une méthode exacte (déterministe) dans un délai " raisonnable ". Ces problèmes peuvent être divisés en plusieurs catégories selon qu'ils sont continus ou discrets, contraints ou non, mono ou multi-objectifs, statiques ou dynamiques.

Dans la recherche d'optimalité, il y a des buts à atteindre et des contraintes à satisfaire [Faturechi et Miller-Hooks, 2014, Aldwaik et Adeli, 2014], le but fixé est la fonction objective qui varie d'une forme de problème à l'autre [Gao et Zhang, 2013, Zhang et Wang, 2013].

Il est important de noter que le but de l'optimisation dans tout système de contraintes est de trouver des valeurs pour un paramètre qui optimisera un objectif établi. L'approche classique ou la recherche heuristique locale peut être utilisée à la recherche d'une solution réalisable ou quasi-optimale. Par conséquent, de nombreux chercheurs ont commencé à proposer des nouvelles techniques de résolution appelées les algorithmes métaheuristiques , ces algorithmes sont utilisés pour résoudre des problèmes d'optimisation complexes dans un temps et un coût de calcul raisonnable. Au cours des dernières décennies, les métaheuristiques sont devenues plus populaires parmi les chercheurs, car, elles présentent plusieurs avantages par rapport aux techniques traditionnelles. Le premier et le plus important avantage de l'algorithme métaheuristique par rapport aux méthodes d'optimisation traditionnelles est qu'il n'a pas besoin d'informations de gradient pour résoudre les problèmes d'optimisation. Le second est qu'ils sont simples et faciles à mettre en œuvre dans différents domaines d'études. le troisième, en utilisant leur capacité d'exploration, ils peuvent réduire la probabilité de piégeage dans les optima locaux.

## 1.2 PROBLÈME D'OPTIMISATION

Un problème d'optimisation est défini comme la recherche dans un espace de solutions, d'une solution optimale quantifiée par une fonction objective. Cette quantification conduit à vouloir maximiser ou minimiser le problème. Un ensemble de méthodes exactes permet de trouver une solution en un temps fini et de manière générale, polynomial. Ces méthodes, parmi lesquelles on trouve la méthode du gradient, la méthode de Newton, etc., ces méthodes sont des méthodes déterministes c'est à dire qu'elles fournissent le meilleur résultat attendu en un nombre fini d'étapes, par dérivation ou par parcours de tout ou une partie des solutions. L'optimisation difficile représente une classe de problèmes d'optimisation qui ne peut être résolue en un temps polynomial ou par une méthode exacte, sous la contrainte des caractéristiques de la fonction objective.

### 1.2.1 Formulation du problème d'optimisation

Mathématiquement, il est possible d'écrire la plupart des problèmes d'optimisation sous la forme générique à minimiser par la fonction suivante :

$$f_i(x), (i = 1, 2, 3, \dots, M), x \in R^D \quad (1.1)$$

avec :

$$h_j(x) = 0, (j = 1, 2, 3, \dots, J) \quad (1.2)$$

$$g_k(x) \leq 0, (k = 1, 2, 3, \dots, K) \quad (1.3)$$

Où  $f_i(x), h_j(x), g_k(x)$  sont des fonctions du vecteur de conception. elles représentent les composantes du vecteur de dimension  $D$ ,  $x$  est le variable de conception ou de décision réelles, continues, discrètes. La fonction  $f_i(x)$ , où, ( $i = 1, 2, 3, \dots, M$ ) est appelée fonction objective ou simplement fonction de coût, et dans le cas de  $M = 1$ , il n'y a qu'un seul objectif. L'espace couvert par les variables de décision est appelé espace de conception ou espace de recherche  $R^D$ , tandis que l'espace formé par les valeurs de la fonction objective est appelé espace de solution ou espace de réponse. Les égalités pour  $h_j(x)$  et les inégalités pour  $g_k$  sont appelées des fonctions de contraintes. Il convient de souligner que nous pouvons également écrire les inégalités dans l'autre sens,  $\geq 0$ , et que nous pouvons également formuler les objectifs comme un problème de maximisation. Dans un cas rare, mais extrême où il n'y a pas du tout d'objectif, il n'y a que des contraintes. Un tel problème s'appelle un problème de faisabilité, car toute solution réalisable est une solution optimale.

### 1.2.2 Classification des problèmes d'optimisation

La classification des problèmes d'optimisation peut également être effectuée en examinant la modalité du paysage objectif, qui peut être divisée en problèmes multi-modaux et problèmes uni-modaux (y compris l'optimisation convexe). En outre, la classification peut également concerner la détermination du problème. S'il n'y a pas de hasard dans la formulation,

le problème est dit déterministe et tous ces problèmes antérieurs sont essentiellement déterministes. Cependant, s'il y a une incertitude dans les variables ou les formes de fonctions, alors une telle optimisation implique des distributions de probabilité et des attentes. Dans ce cas, ils sont souvent appelés optimisation stochastique ou optimisation robuste. La classification des problèmes d'optimisation et les terminologies utilisées dans la littérature sur l'optimisation qui peut être diverses et potentiellement déroutantes. Le fait qu'un problème d'optimisation soit considéré comme facile ou difficile peut dépendre de nombreux facteurs et de la formulation mathématique réelle ou des fonctions du problème. En fait, trois facteurs qui rendent un problème plus difficile sont : la non-linéarité de la fonction objective, la dimensionnalité élevée du problème et la forme complexe du domaine de recherche.

- **Non-linéarité et multi-modalité :**

La non-linéarité élevée d'une fonction à optimiser signifie généralement une multi-modalité avec plusieurs modes locaux avec des optima multiples (voire infinis). Dans la plupart des cas, les algorithmes pour résoudre de tels problèmes sont plus susceptibles d'être piégés dans les modes locaux.

- **Haute dimensionnalité :**

La haute dimensionnalité est associée à la malédiction de la dimensionnalité où la distance devient grande, alors que le volume de tout algorithme qui peut réellement rechercher devient insignifiant par rapport au vaste espace réalisable.

- **Contraintes :**

Des contraintes multiples peuvent compliquer la région réalisable avec une géométrie ou des formes irrégulières. Dans certains cas, les régions réalisables peuvent être divisées en plusieurs régions déconnectées avec des îles isolées, ce qui rend plus difficile pour les algorithmes de rechercher toutes les régions réalisables (ainsi potentiellement manquer la véritable optimalité).

D'autres facteurs tels que le temps d'évaluation d'un objectif qui est également important. Dans de nombreuses applications, telles que la bioinformatique, l'ingénierie aérospatiale et l'apprentissage automatique en profondeur, l'évaluation d'un seul objectif peut prendre beaucoup de temps (de quelques heures à quelques jours, voire des semaines), par conséquent les coûts de calcul peuvent être très élevés. Ainsi, le choix d'algorithmes efficaces devient primordial.

### 1.3 LES MÉTAHEURISTIQUES D'OPTIMISATION

Les méthodes d'optimisation visent à trouver une solution optimale ou quasi optimale à un problème et peuvent être divisées en méthodes

exactes et (méta) heuristiques. Les méthodes exactes sont conçues de manière à permettre de trouver une solution optimale dans un laps de temps limité, tandis que les méthodes (méta) heuristiques ne peuvent pas garantir l'optimalité. Cependant, les méthodes exactes prennent souvent trop de temps (temps de calcul), en particulier en présence de problèmes combinatoires de dimension réaliste, car une énumération explicite ou implicite est effectuée. Ainsi, à moins que les exigences de temps ne croissent pas de manière polynomiale avec la taille du problème ou que les instances à résoudre soient de petite taille, une méthode méta-heuristique devrait être utilisée. Les méthodes heuristiques et les méthodes métaheuristiques diffèrent en ce sens que tandis que les premiers exploitent les propriétés des problèmes, les seconds sont généraux et peuvent être appliqués à n'importe quel problème. Ainsi, les méthodes heuristiques sont conçues pour un problème spécifique et incorporent ses caractéristiques. Ce faisant, des solutions réalisables et assez souvent raisonnablement bonnes sont trouvées très rapidement, néanmoins, ils ont une tendance à être trop gourmands et donc à se retrouver piégés dans des optima locaux. La métaheuristique fournit un cadre général qui peut être utilisé pour résoudre différents problèmes, comme une boîte noire. En outre, ils comprennent également un module spécifique au problème, souvent appelé heuristique, qui permet une implémentation spécifique au problème du cadre susmentionné.

Aujourd'hui, les métaheuristiques jouent un rôle important dans la résolution des problèmes d'optimisation où l'optimisation mathématique et irréalizable soit en raison du temps de calcul long ou de définitions incomplètes des problèmes. Ils utilisent une combinaison de méthodes heuristiques, disposées dans un cadre de niveau supérieur, pour fournir des capacités d'exploration de l'espace de recherches plus efficaces [Blum et Roli, 2003a] et peuvent être décrites comme un modèle de conception, plutôt qu'un algorithme [Lones, 2014]. Cependant, un métaheuristique ne peut pas effectuer aussi bien sûr tous les problèmes d'optimisation que le prouvent les théorèmes d'optimisation No Free Lunch [Wolpert et Macready, 1997]. Ces facteurs et d'autres ont conduit à une explosion du nombre de métaheuristiques inspirées de la nature au cours des dernières années.

### 1.3.1 Concepts des techniques métaheuristiques

Le mot "heuristique" est défini dans le contexte de l'informatique comme une méthode permettant de désigner une règle empirique permettant de résoudre un problème sans l'application exhaustive d'une procédure. En d'autres termes, une heuristique est une méthode qui :

- (i) recherche une solution approximative.
- (ii) n'a pas particulièrement besoin d'une preuve mathématique de convergence.
- (iii) n'explore pas toutes les solutions possibles dans l'espace de recherche avant d'arriver à la solution finale, est donc efficace en matière de calcul. Une méthode métaheuristique est particulièrement pertinente dans le cadre de la résolution de problèmes de recherche et d'optimisation. Il décrit une méthode qui utilise une ou plusieurs heuristiques et hérite donc

des trois propriétés mentionnées ci-dessus. Ainsi, une méthode métaheuristique permet :

- (i) cherche à trouver une solution quasi optimale, au lieu d'essayer spécifiquement de trouver la solution optimale exacte.
- (ii) n'a généralement pas de preuve rigoureuse de convergence vers la solution optimale.
- (iii) est généralement plus rapide du point de vue du calcul que la recherche exhaustive.

Ces méthodes sont de nature itérative et utilisent souvent des opérations stochastiques dans leur processus de recherche pour modifier une ou plusieurs solutions candidates initiales . Étant donné que de nombreux problèmes d'optimisation du monde réel sont complexes en raison de leurs aspects pratiques inhérents, les algorithmes d'optimisation classiques peuvent ne pas toujours être applicables et peuvent ne pas convenir à la résolution de tels problèmes de manière pragmatique. Conscients de ce fait et sans négliger l'importance des algorithmes classiques dans le développement du domaine de la recherche et de l'optimisation, les chercheurs et les praticiens ont recherché des méthodes métaheuristiques afin qu'une solution quasi optimale puisse être obtenue de manière calculable, au lieu d'attendre un algorithme d'optimisation prouvable à développer avant de tenter de résoudre de tels problèmes. La capacité des méthodes métaheuristiques à gérer différentes complexités associées à des problèmes pratiques et à arriver à une solution raisonnablement acceptable est la principale raison de la popularité des méthodes métaheuristiques dans un passé récent. La plupart des méthodes métaheuristiques sont motivées par des principes naturels, physiques ou biologiques et tentent de les imiter à un niveau fondamental à travers divers opérateurs. Un thème commun à toutes les métaheuristiques est l'équilibre entre l'exploration et l'exploitation. L'exploration fait référence à la manière dont les opérateurs diversifient les solutions dans l'espace de recherche. Cet aspect donne au métaheuristique un comportement de recherche global. Tandis que l'exploitation fait référence à la capacité des opérateurs à utiliser les informations disponibles à partir des solutions des itérations précédentes pour intensifier la recherche. Une telle intensification donne au métaheuristique une caractéristique de recherche locale. Certaines métaheuristiques ont tendance à être plus exploratoires qu'exploitantes, tandis que d'autres font le contraire. Par exemple, la méthode primitive de sélection aléatoire de solutions pour un certain nombre d'itérations, représente une recherche complètement exploratoire. D'autre part, les métaheuristiques permettent d'ajuster l'équilibre entre diversification et intensification par l'utilisateur au travers de paramètres opérateurs.

### 1.3.2 Classification des techniques métaheuristiques

Il existe plusieurs façons de classer les métaheuristiques. On en donne quelques une, et nous adopterons celle faisant la différence entre les méthodes de trajectoire et les méthodes basées sur une population. La manière la plus courante de classer les techniques métaheuristiques est basée sur le nombre de solutions initiales qui sont modifiées dans les itérations suivantes. Les métaheuristiques à solution unique commencent par une

solution initiale qui est modifiée de manière itérative. Notez que le processus de modification lui-même peut impliquer plus d'une solution, mais qu'une seule solution est utilisée dans chaque itération suivante. Les métaheuristiques basées sur la population utilisent plus d'une solution initiale pour démarrer l'optimisation. À chaque itération, plusieurs solutions sont modifiées, et certains d'entre eux atteignent la prochaine itération. La modification des solutions se fait via des opérateurs qui utilisent souvent des propriétés statistiques particulières de la population. Le paramètre supplémentaire pour la taille de la population est définie par l'utilisateur et reste généralement constant à travers les itérations. Une autre façon de classer les métaheuristiques consiste à utiliser le domaine qu'elles imitent. Des termes génériques tels que bio-inspiré et inspiré de la nature sont souvent utilisés pour les métaheuristiques. Cependant, ils peuvent être sous-catégorisés en tant qu'algorithmes évolutifs, algorithmes basés sur l'intelligence en essaim et algorithmes basés sur des phénomènes physiques. Les algorithmes évolutifs (comme les algorithmes génétiques, les stratégies d'évolution, l'évolution différentielle, la programmation génétique, la programmation évolutive, etc.) imitent divers aspects de l'évolution dans la nature la reproduction et la mutation génétique. Les algorithmes de Swarm-Intelligence imitent le comportement de groupe et/ou les interactions des organismes vivants (comme les fourmis, les abeilles, les oiseaux, les lucioles, les poissons, les globules blancs, les bactéries, etc.) et les êtres non vivants (comme les gouttes d'eau, etc.). Le reste de la métaheuristique imite divers phénomènes physiques comme le recuit des métaux, l'esthétique musicale (harmonie), etc. Une quatrième sous-catégorie peut être utilisée pour classer les métaheuristiques dont la source d'inspiration n'est pas claire (comme la recherche tabou et la recherche par dispersion) ou celles qui sont trop peu nombreuses pour avoir une catégorie pour eux-mêmes. D'autres méthodes populaires de classification des métaheuristiques sont [Talbi, 2009, Birattari et al., 2001] :

#### **Méthodes déterministes ou stochastiques :**

Les méthodes déterministes suivent une trajectoire définie à partir de la ou des solutions initiales aléatoires. Par conséquent, elles sont parfois appelées méthodes de trajectoire. Les méthodes stochastiques (également les méthodes discontinues) permettent des sauts probabilistes de la ou des solutions actuelles à la suivante.

#### **Méthodes gourmandes ou non gourmandes :**

Les algorithmes gourmands recherchent généralement dans le voisinage de la solution actuelle et se déplacent immédiatement vers une meilleure solution lorsqu'elle est trouvée. Ce comportement souvent conduit à un optimum local. Les méthodes non gourmandes tiennent certaines itérations avant de mettre à jour la ou les solutions ou disposent d'un mécanisme pour revenir en arrière à partir d'un optimum local. Cependant, pour les problèmes convexes, un comportement gourmand est la stratégie optimale.

**Utilisation de la mémoire par rapport aux méthodes sans mémoire :**

Les méthodes basées sur la mémoire conservent un enregistrement des solutions passées et de leurs trajectoires et les utilisent pour diriger la recherche. Une méthode de mémoire populaire appelée " la recherche tabou " .

**Une contre plusieurs méthodes de voisinage :**

Certaines métaheuristiques comme le recuit simulé et la recherche tabou ne permettent qu'un ensemble limité de mouvements à partir de la solution actuelle. Mais de nombreuses métaheuristiques utilisent des opérateurs et des paramètres pour autoriser plusieurs quartiers. Par exemple, l'optimisation des essaims de particules y parvient grâce à diverses topologies d'essaims.

**Fonction objectif dynamique ou statique :**

Les métaheuristiques qui mettent à jour la fonction objective en fonction des exigences actuelles de la recherche sont classées comme dynamiques. D'autres métaheuristiques utilisent simplement leurs opérateurs pour contrôler la recherche.

**1.4 LES CARACTÉRISTIQUES DES MÉTAHEURISTIQUES :**

1 - Les métaheuristiques sont des stratégies qui permettent de guider la recherche d'une solution.

2 - Le but visé par les métaheuristiques est d'explorer l'espace de recherche efficacement afin de déterminer des points (presque) optimaux.

3 - Les techniques qui constituent des algorithmes du type métaheuristique vont de la simple procédure de recherche locale à des processus d'apprentissage complexes.

4 - Les métaheuristiques sont en général non déterministes et ne donnent aucune garantie d'optimalité.

5 - Les métaheuristiques peuvent contenir des mécanismes qui permettent d'éviter d'être bloqué dans des régions de l'espace de recherche.

6 - Les concepts de base des métaheuristiques peuvent être décrits de manière abstraite, sans faire appel à un problème spécifique.

7 - Les métaheuristiques peuvent faire appel à des heuristiques qui tiennent compte de la spécificité du problème traité, mais ces heuristiques sont contrôlées par une stratégie de niveau supérieur.

8 - Les métaheuristiques peuvent faire usage de l'expérience accumulée durant la recherche de l'optimum, pour mieux guider la suite du processus de recherche.

9 - La métaheuristique peut conduire à des solutions suffisamment bonnes pour des problèmes de calcul (techniquement, classe P) simples avec une grande complexité d'entrée, ce qui peut être un obstacle pour les méthodes classiques.

10 - La métaheuristique peut conduire à des solutions suffisamment bonnes pour les problèmes NP-difficiles, c'est-à-dire des problèmes pour

lesquels il n'existe aucun algorithme exact connu qui puisse les résoudre dans un laps de temps raisonnable.

11 - Contrairement à la plupart des méthodes classiques, la métaheuristique ne nécessite aucune information sur le gradient et peut donc être utilisée avec des fonctions objectives non analytiques, en boîte noire ou basée sur la simulation.

12 - La plupart des métaheuristicues ont la capacité de récupérer des optima locaux en raison de la stochasticité inhérente ou des heuristiques déterministes spécifiquement destinées à cet effet

13 - En raison de la capacité de récupérer des optima locaux, les métaheuristicues peuvent mieux gérer les incertitudes dans les objectifs.

14 - La plupart des métaheuristicues peuvent gérer plusieurs objectifs avec seulement quelques changements algorithmiques.

## 1.5 L'INSPIRATION DES MÉTAHEURISTIQUES :

Le concept de base de la plupart des algorithmes métaheuristicues s'inspire de la nature, du comportement animal ou des phénomènes physiques [Khalilpourazari et Khalilpourazary, 2016, Mirjalili et Lewis, 2016]. les algorithmes métaheuristicues sont classés en trois classes principales : techniques basées sur l'évolution, basées sur la physique et basées sur les essais. Les méthodes évolutives imitent le processus d'évolution dans la nature pour effectuer l'optimisation [Khalilpourazari et Khamseh, 2019]. Alors que les algorithmes basés sur la physique effectuent une optimisation en utilisant les règles de la physique dans l'univers. La troisième classe des algorithmes métaheuristicues sont des techniques basées sur Swarm, qui imitent le comportement des animaux dans un groupe.

Les méthodes d'optimisation métaheuristique inspirées de la nature ont conduit à la découverte d'un grand nombre d'algorithmes d'optimisation dans diverses catégories, comme détaillé ci-dessous :

### 1.5.1 Algorithmes évolutifs :

Les algorithmes évolutifs utilisent généralement une procédure itérative, basée sur une progression de l'évolution biologique pour résoudre des problèmes d'optimisation. Certains des algorithmes évolutifs sont décrits ci-dessous :

#### Evolution différentielle

L'évolution différentielle (DE) été développée par R. Storn et K. Price dans [Storn et Price, 1997]. DE est un algorithme métaheuristique à base de vecteurs, qui présente une certaine similitude avec les algorithmes génétiques en raison de son utilisation du croisement et de la mutation. En fait, l'algorithme DE peut être considérée comme un développement ultérieur des algorithmes génétiques avec des équations de mise à jour explicites qui permettent de faire une analyse théorique. DE est un algorithme de recherche stochastique à tendance auto-organisée et n'utilise pas les informations de dérivées. Ainsi, il s'agit d'un système basé sur la population. De plus, l'algorithme DE utilise des nombres réels comme une chaînes de

solution, donc aucun codage ni décodage n'est nécessaire. Comme dans les algorithmes génétiques, les paramètres de conception dans un espace de recherche sont représentés sous forme de vecteurs, et divers opérateurs génétiques sont opérés sur leurs bits de chaînes. Cependant, contrairement aux algorithmes génétiques, l'algorithme DE effectue des opérations sur chaque composant (ou chaque dimension de la solution). Presque tout est fait en matière de vecteurs. Par exemple, dans les algorithmes génétiques, la mutation est effectuée sur un ou plusieurs sites d'un chromosome, alors qu'en évolution différentielle, un vecteur de différence de deux vecteurs de populations choisies au hasard est utilisé pour perturber un vecteur existant. Une telle mutation vectorisée peut être considérée comme une approche plus efficace du point de vue de la mise en œuvre. Ce type de perturbation est effectué sur chaque vecteur de population et on peut donc s'attendre à ce qu'il soit plus efficace. De même, le croisement (crossover) est également un échange de composants de chromosomes ou de segments de vecteur basé sur des vecteurs. d'autre utilisation de la mutation et du croisement comme opérateurs différentiels, DE a des équations de mise à jour explicites. Cela facilite également la mise en œuvre et la conception de nouvelles variantes

Pour un problème d'optimisation de " $d$ " paramètres, une population de " $n$ " solution, les vecteurs sont initialement générés. Nous avons  $x_i$ , où  $i = 1, 2, \dots, n$ . Pour chaque solution  $x_i$  à n'importe quelle génération  $t$ , nous utilisons la notation conventionnelle définie par l'équation suivante :

$$x_i^t = (x_{1,i}^t, x_{2,i}^t, x_{3,i}^t, \dots, x_{d,i}^t) \quad (1.4)$$

L'évolution différentielle comprend trois étapes principales : la mutation, le croisement et la sélection. La mutation est réalisée par le schéma de mutation. Pour chaque vecteur  $x_i$  à tout moment ou génération  $t$ , nous choisissons d'abord au hasard trois vecteurs distincts  $x_p$ ,  $x_q$  et  $x_r$  en  $t$ , puis nous générons un vecteur dit donneur par le schéma de mutation :

$$v_i^{t+1} = x_p^t + F(x_k^t - x_r^t) \quad (1.5)$$

où  $F \in [0, 2]$  est un paramètre appelé poids différentiel. Cela nécessite que le nombre minimum de la taille de la population soit  $n \geq 4$ . En principe,  $F \in [0, 2]$ , mais en pratique, un schéma avec  $F \in [0, 1]$  est plus efficace et stable. En fait, presque toutes les études de la littérature utilisent  $F \in (0, 1)$ . Le croisement est contrôlé par un paramètre de croisement  $C_r \in [0, 1]$ , contrôlant le taux ou la probabilité de croisement. Le croisement proprement peut être effectué de deux manières : binomial et exponentiel. Le schéma binomial effectue un croisement sur chacun des  $d$  composants ou variables/paramètres. En générant un nombre aléatoire uniformément distribué  $r_i \in [0, 1]$ , la  $j^{me}$  composante de  $v_i$  est manipulée par :

$$u_{j,i}^{t+1} = \begin{cases} v_{j,i} & \text{si } r_i \leq C_r \\ x_{j,i}^t & \text{sinon } j = 1, 2, \dots, d \end{cases} \quad (1.6)$$

De cette façon, il peut être décidé au hasard s'il faut ou non échanger chaque composant avec un vecteur donneur. Dans le schéma exponentiel, un segment du vecteur donneur est sélectionné et ce segment commence

par un entier aléatoire  $k$  avec une longueur aléatoire  $L$ . Mathématiquement, il s'agit de choisir aléatoirement  $k \in [0, d - 1]$  et  $L \in [1, d]$ , d'où :

$$u_{j,i}^{t+1} = \begin{cases} v_{j,i}^t & \text{si } j = k, \dots, k - L + 1 \in [1, d] \\ x_{j,i}^t & \text{sinon} \end{cases} \quad (1.7)$$

nous utiliserons le croisement binomial dans notre implémentation. La sélection est essentiellement la même que celle utilisée dans les algorithmes génétiques. Nous sélectionnons la valeur la plus adaptée pour le problème de minimisation, la valeur minimale de l'objectif. Par conséquent, nous avons :

$$x_i^{t+1} = \begin{cases} u_i^{t+1} & \text{si } f(u_i^{t+1}) \leq f(x_i^t) \\ x_i^t & \text{sinon} \end{cases} \quad (1.8)$$

L'efficacité globale de la recherche est contrôlée par deux paramètres : le poids différentiel  $F$  et la probabilité de croisement  $C_r$ .

### L'algorithme génétique

L'algorithme génétique est une technique évolutionnaire proposé dans [Holland et Reitman, 1977]. Cet algorithme est basée sur trois étapes principales :

**1- L'étape de sélection :** pour déterminer les meilleures solutions, qui sont utilisées pour engendrer la nouvelle génération. Il y a différentes stratégies de sélection comme l'algorithme de sélection de roue de roulette (roulette-wheel selection technique), la sélection par classement (ranking selection), la sélection par tournoi (tournament selection), etc, ...

**2- L'étape de croisement :** qui combine les caractéristiques des parents (préalablement sélectionnés) pour générer les enfants. Un exemple de croisement est illustré sur la figure.1.1.

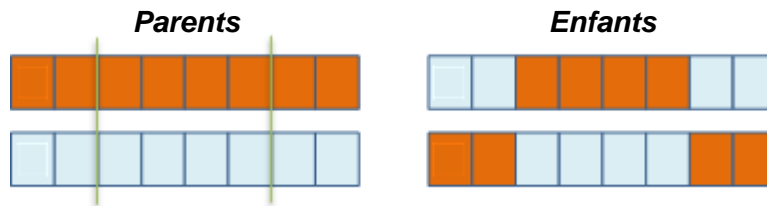


FIGURE 1.1 – La phase de croisement .

**3- L'étape de mutation :** qui modifie aléatoirement une partie de l'individu, ce qui permet de maintenir une certaine diversité dans la population. Un exemple de la mutation est illustré sur la figure.1.2

Après ces trois étapes, certains individus parents sont remplacés par d'autres individus enfants. La stratégie de remplacement la plus simple est de prendre les meilleurs individus de la population en fonction de leurs performances respectives.

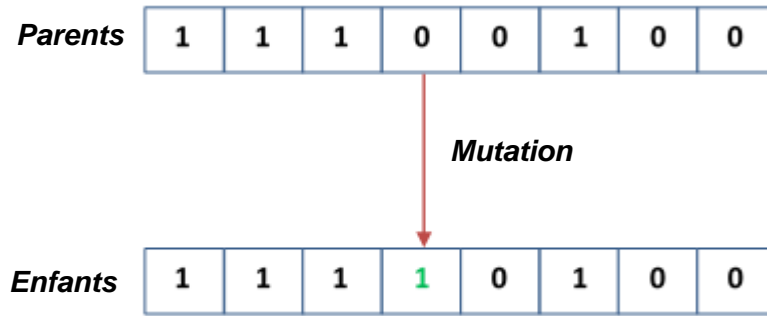


FIGURE 1.2 – La phase de mutation .

### 1.5.2 Algorithmes basés sur Swarm Intelligence :

Les algorithmes basés sur l'intelligence des essaims utilisent le comportement collectif d'animaux tels que les oiseaux, les insectes ou les poissons. Ici, nous présentons brièvement certains des algorithmes d'essaim les plus largement utilisés.

#### Algorithme d'essaim particules :

L'algorithme d'essaim particules recherche l'espace d'une fonction objective en ajustant les trajectoires d'agents individuels, appelés particules, comme les chemins par morceaux formés par des vecteurs positionnels de manière quasi-stochastique [Kennedy et Eberhart, 1995, Dorigo et Bittari, 2007, Engelbrecht, 2005]. Le mouvement d'une particule en essaim se compose de deux composants principaux : un composant stochastique et un composant déterministe. Chaque particule est attirée vers la position du meilleur solution global actuel  $g^*$  et son propre meilleur emplacement  $x_i^*$ , alors qu'en même temps, elle a tendance à se déplacer de manière aléatoire. Lorsqu'une particule trouve un emplacement meilleur que tous les emplacements précédemment trouvés, il met à jour cet emplacement comme le nouveau meilleur courant pour la particule  $i$ . Il existe un meilleur emplacement courant pour toutes les  $n$  particules à tout moment  $t$  pendant les itérations. L'objectif est de trouver le meilleur global parmi toutes les meilleures solutions actuelles jusqu'à ce que l'objectif ne s'améliore plus ou après un certain nombre d'itérations. Le mouvement des particules est schématiquement représenté sur la figure 1.3.

Où  $x_i^{*(t)}$  représente le meilleur emplacement courant pour la particule  $i$ , et  $g^* \approx \min f(x_i)$  pour  $(i = 1, 2, \dots, n)$  est le meilleur courant global en  $t$ .

Alors,  $x_i$  représente le vecteur de position et  $v_i$  la vitesse de la particule  $i$ . Le nouveau vecteur de vitesse est déterminé par la formule suivante :

$$v_i^{t+1} = v_i^t + \alpha \epsilon_1 [g^* - x_i^t] + \beta \epsilon_2 [x_i^{*(t)} - x_i^t] \quad (1.9)$$

D'où  $\epsilon_1$  et  $\epsilon_2$  sont deux vecteurs aléatoires, et chaque entrée prend les valeurs comprises entre 0 et 1. Les paramètres  $\alpha$  et  $\beta$  sont les paramètres d'apprentissage ou les constantes d'accélération qui peuvent généralement être pris proche de  $\approx 2$ . Les emplacements initiaux de toutes les particules doivent être distribués de manière relativement uniforme afin de pouvoir

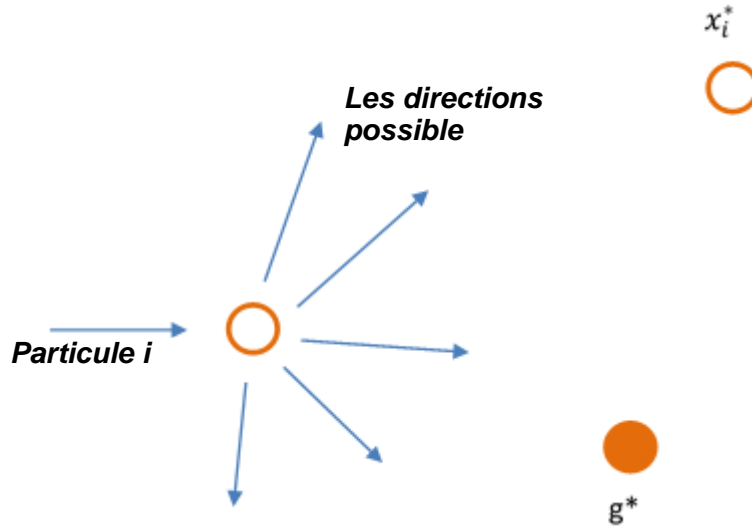


FIGURE 1.3 – Représentation schématique du mouvement d'une particule dans PSO se déplaçant vers le meilleur global  $g^*$  et le meilleur courant  $x_i^*$  pour chaque particule  $i$ .

échantillonner dans la plupart des régions, ce qui est particulièrement important pour les problèmes multi-modaux. La vitesse initiale d'une particule peut être prise égale à zéro, c'est-à-dire  $v_i^{t=0} = 0$ . La nouvelle position peut alors être mise à jour par l'équation suivante :

$$v_i^{t+1} = x_i^t + v_i^{t+1} \quad (1.10)$$

Bien que  $v_i$  puisse être n'importe quelle valeur, il est généralement limité dans une certaine plage  $[0, v_{max}]$ . Il existe de nombreuses variantes qui étendent l'algorithme PSO standard, et l'amélioration la plus notable est probablement d'utiliser la fonction d'inertie  $\theta(t)$  pour que  $v_i^t$  soit remplacé par  $\theta(t) v_i^t$

$$v_i^{t+1} = \theta v_i^t + \alpha \epsilon_1 [g^* - x_i^t] + \beta \epsilon_2 [x_i^{*(k)} - x_i^t] \quad (1.11)$$

Où  $\theta$  prend les valeurs entre 0 et 1 en théorie. Dans le cas le plus simple, la fonction d'inertie peut être prise comme une constante, typiquement  $\theta \approx 0,5 \sim 0,9$ . Cela équivaut à introduire une masse virtuelle pour stabiliser le mouvement des particules et donc l'algorithme devrait converger plus rapidement.

### Algorithme de chauve-souris :

L'algorithme de chauve-souris est un nouvel algorithme métaheuristique basé sur la population développé dans [Yang et Gandomi, 2012], il s'inspire du comportement l'écholocalisation des animaux fascinants appelés chauves-souris, les micro chauves-souris utilisent une écho-localisation pour trouver la proie, localiser leurs crevasses perchées dans l'obscurité et éviter les obstacles, cette micro chauve-souris a la capacité de trouver la proie en utilisant la transmission d'un signal élevé et de signaux audio courts. La capacité d'écholocalisation avancée des chauves-souris les rend fascinantes et fonctionne parfaitement dans un environnement sombre. Les cinq étapes principales de l'algorithme de chauve-souris sont :

### Étape 1 : Initialisation de la population de chauves-souris

Les groupes de la population sont générés aléatoirement pour chaque valeur de dimension "d", les chauves-souris volent avec des valeurs initiales de : l'intensité, la fréquence, la vitesse et le taux d'émission d'impulsions

$$x(i, j) = LL(j) + rand(0, 1)xUL(j) - LL(j) \quad (1.12)$$

Où  $rand \in [0, 1]$  est un vecteur aléatoire tiré d'une distribution uniforme,  $i = 1, 2, \dots, N$ ,  $j = 1, 2, \dots, d$ , la valeur de la limite supérieure (UL) et inférieure (LL) dépend de la zone de réseau  $(X_i, Y_i)$ .

### Étape 2 : Mouvement des chauves-souris (mise à jour : fréquence, vitesse et position)

Nous devons utiliser des chauves-souris virtuelles. On définit l'ensemble des règles qui permettent de mettre à jour les positions  $x_i$  et les vitesses  $v_i$  dans un espace de recherche de dimension  $d$ . Les nouvelles solutions  $x_i^t$  et les vitesses  $v_i^t$  à l'instant  $t$  sont définies comme suit :

$$f_i = f_{min} + f_{(max - f_{min})} * rand \quad (1.13)$$

$$v_i^{t+1} = v_i^t + (x_i^t - x_*) * f_i \quad (1.14)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (1.15)$$

où  $f_i \in [f_{min}, f_{max}]$  est la fréquence,  $rand \in [0, 1]$  est un vecteur aléatoire tiré d'une distribution uniforme, et  $x_*$  le meilleur emplacement global actuel (solution),

### Étape 3 : Recherche locale

Pour cette partie, une fois qu'une solution est sélectionnée parmi les meilleures solutions actuelles, une nouvelle solution pour chaque chauve-souris est générée localement en utilisant une trajectoire aléatoire :

$$if(rand(0, 1) > r_i \quad (1.16)$$

$$x_{new} = x_{old} + \epsilon * A^t \quad (1.17)$$

Où  $\epsilon \in [-1, 1]$  est une valeur aléatoire,  $A^t$  représente l'intensité moyenne des chauves-souris à l'instant  $t$ ,  $r_i \in [0, 1]$  est le rapport d'impulsion. Lorsque la meilleure valeur obtenue est supérieure à la fonction précédente  $f(x_*)$ , alors, la meilleure solution globale  $x_*$  peut être mise à jour.

### Étape 4 : mise à jour l'intensité et l'émission d'impulsions

En outre, l'intensité  $A_i$  et le taux d'émission d'impulsions  $r_i$  doivent être mis à jour en conséquence au fur et à mesure des itérations. Parce que le volume d'intensité diminue généralement une fois qu'une chauve-souris a trouvé sa proie, alors que le taux d'émission d'impulsions augmente, Pour simplification, la valeur de  $A_0 = 1$  et  $A_{min} = 0$ , ainsi, en

supposant que  $A_{min} = 0$  signifie qu'une chauve-souris vient de trouver la proie et arrête temporairement d'émettre le signal. Cette relation peut être expliquée comme suit :

$$if(rand(0,1) < A_i^t \&\& f(x_i) < f(x)) \quad (1.18)$$

$$f(x) = f(x_i) \quad (1.19)$$

$$A_i^{t+1} = \alpha * A_i^t \quad (1.20)$$

$$r_i^{t+1} = r_i^0 * (1 - e^{-yt}) \quad (1.21)$$

où  $\alpha \in [0,1]$  et  $y > 0$  sont des constantes,  $r_i^0$  est une valeur aléatoire,  $r_i^0 \in [0,1]$ ,  $A_i^t \in [1,2]$  à  $t = 0$ .

### 1.5.3 Algorithmes d'optimisation basés sur la physique

Les algorithmes d'optimisation basés sur la physique où les phénomènes basés sur la physique sont étudiés et modélisés. Par exemple, le recuit simulé [Bertsimas et Tsitsiklis, 1993], inspiré de la thermodynamique statistique pour le recuit des solides, s'est avéré très utile pour résoudre des problèmes d'optimisation combinatoire complexes.

#### Recuit simulé

Le recuit simulé (SA) est l'une des métaheuristiques largement utilisées, elle est également considérée comme l'une des études les plus étudiées en matière d'analyse de convergence [Bertsimas et Tsitsiklis, 1993, Kirkpatrick et al., 1983].

Le recuit simulé adopte une marche aléatoire basée sur la trajectoire d'un seul agent, à partir d'une estimation initiale  $x_0$ . la prochaine étape est reliée seulement à l'état ou l'emplacement actuel et la probabilité d'acceptation  $p$ . Il s'agit principalement d'une chaîne de Markov dont la probabilité de transition de l'état actuel à l'état suivant est donnée par l'équation suivante :

$$P = \exp\left[\frac{\Delta E}{k_B T}\right] \quad (1.22)$$

Où  $k_B$  est la constante de Boltzmann et  $T$  est la température. Le changement d'énergie  $\Delta E$  peut être lié au changement des valeurs objectives. Des recherches sur la convergence du recuit simulé ont ouvert la voie à l'analyse de tous les algorithmes basés sur le recuit simulé [Bertsimas et Tsitsiklis, 1993, Granville et al., 1994, Steinhöfel et al., 2000]. des chercheurs ont fourni un excellent examen de la convergence sous diverses hypothèses [Bertsimas et Tsitsiklis, 1993, Granville et al., 1994]. En se basant sur les hypothèses dont lequel l'algorithme de recuit simulé établissait comme une chaîne de Markov inhomogène avec des états finis, ils ont prouvé une fonction de convergence probabiliste  $P$ , plutôt qu'une convergence presque sûre, que :

$$\max P[x_i t \in S_* | x_0] \geq \frac{A}{t^\alpha} \quad (1.23)$$

Où  $S^*$  représente l'ensemble optimal,  $A$  et  $\alpha$  sont des constantes positives [Bertsimas et Tsitsiklis, 1993]. C'est pour le programme de refroidissement  $T(t) = d/\ln(t)$ , où  $t$  représente le compteur d'itérations. Ces études ont largement utilisé les chaînes de Markov comme outil principal.

TABLE 1.1 – Inspiration des algorithmes métaheuristiques

Algorithmes évolutifs	Algorithmes basés sur la physique	Algorithmes basés sur l'essaim	Autres algorithmes à base de population
Algorithmes génétiques (GA)[Holland, 1992]	Recuit simulé (SA) [Bertsimas et Tsitsiklis, 1993]	Optimisation des essaims de particules (PSO) [Poli et al., 2007]	Recherche fractale stochastique (SFS) [Salimi, 2015]
Stratégie d'évolution (ES)[L, 1978]	Algorithme de recherche gravitationnelle (GSA) [Rashedi et al., 2009]	Optimiseur du loup gris (GWO) [Mirjalili et Lewis, 2014]	Algorithme sinusoidal (SCA) [Mirjalili, 2016]
Programmation génétique (GP) [Koza, 1993]	Recherche système facturée (CSS)[Kaveh et Talatahari, 2010]	Algorithme de libellule (DA) [Mirjalili, 2015b]	Algorithme du cycle de l'eau (WCA) [Eskandar et al., 2012]
Optimiseur basé sur la biogéographie [Simon, 2008]	Optimisation de la force centrale (CFO) [Formato, 2007]	Optimisation de la flamme des mites (MFO) [Mirjalili, 2015c]	Optimisation des échanges thermiques [Kaveh et Dadras, 2017]
Programmation évolutive (EP) [Fogel et al., 1966]	Algorithme Black Hole (BH) [Hatamlou, 2013]	Algorithme d'optimisation des baleines (WOA) [Mirjalili et Lewis, 2016]	Recherche de colonies virales [Li et al., 2016a]
	Algorithme de recherche basé sur la galaxie (GBSA) [Kaveh et Khayatazad, 2012]	Colonie d'abeilles artificielle (ABC) [Akay et Karaboga, 2012]	
	Algorithme d'optimisation des petits mondes (SWOA) [Du et al., 2006]	Algorithme d'optimisation Ant Lion (ALO) [Mirjalili, 2015a]	
	Optimisation de l'espace courbe (CSO) [Moghadam et al., 2012]	Recherche de coucou (CS) [Yang et Deb, 2009]	
	Algorithme d'optimisation multi-versets (MVO)[Mirjalili et Hatamlou, 2015]	Algorithme dynamique des chauves-souris virtuelles [Topal et Altun, 2016]	

## 1.6 MOTIVATIONS DE L'HYBRIDATION DES MÉTAHEURISTIQUES

Dans un algorithme hybride, deux algorithmes ou plus résolvent collectivement et en coopération un problème prédéfini. Dans certains types d'hybridation, un algorithme peut être incorporé en tant que sous-algorithme pour localiser les paramètres optimaux pour un autre algorithme, tandis que dans d'autres cas, différents composants d'algorithmes tels que la mutation et le croisement sont utilisés pour améliorer un autre algorithme dans la structure hybride. En ce qui concerne cette nature, les algorithmes hybrides peuvent être divisés en deux catégories :

### 1.6.1 Hybridation à usage unique :

Dans cette catégorie, tous les sous-algorithmes sont utilisés pour résoudre directement le même problème, en outre, différents sous-algorithmes sont utilisés dans différentes étapes de recherche. Les algorithmes métaheuristiques hybrides avec recherche locale sont un exemple typique. La recherche globale explore l'espace de recherche, tandis que la recherche locale est utilisée pour affiner les zones qui peuvent contenir l'optimum global.

### 1.6.2 Hybridation à usages multiples :

Dans cette catégorie, un algorithme principal est utilisé pour résoudre le problème, tandis que le sous algorithme est appliqué pour régler les paramètres de l'algorithme principal. A titre d'exemple, l'algorithme d'essaim particule (PSO) peut être appliqué pour trouver la valeur optimale du taux de mutation dans l'algorithme génétique (GA). Par la présente, l'algorithme d'essaim particule ne résout pas le problème, mais principalement aide à trouver de meilleures solutions en recherchant le paramètre optimal pour de meilleures performances. Les algorithmes hyper-heuristiques peuvent être considérés comme une sorte de méthodes hybrides. Dans les méthodes hyper-heuristiques, les paramètres sont sélectionnés (par un sous-algorithme ou via un mécanisme d'apprentissage) [Burke et al., 2013]

## 1.7 TAXONOMIE DES ALGORITHMES HYBRIDES :

De manière générale, les algorithmes hybrides peuvent être regroupés en deux catégories, qui sont décrites dans les sous-sections suivantes.

### 1.7.1 Hybrides collaboratifs :

Cela implique la combinaison de deux algorithmes ou plus s'exécutant en séquence ou en parallèle. Le poids contributif de chaque algorithme participant peut être considéré comme moitié-moitié dans le cas le plus simple. Les structures possibles des algorithmes hybrides dans cette catégorie sont illustrés sur la figure 1.4. Trois structures sont représentés sur cette figure, qui sont :

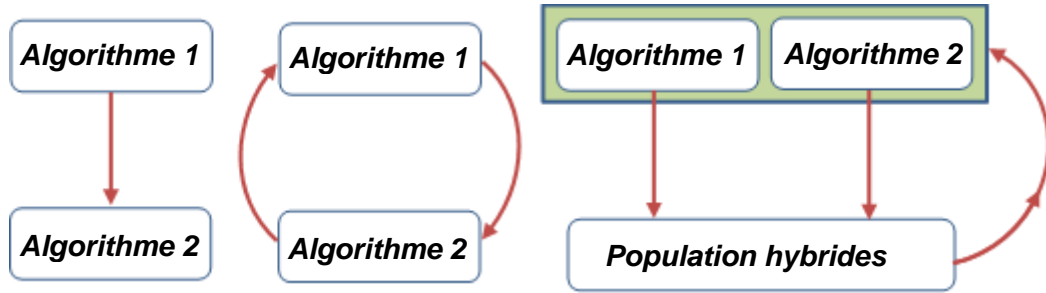


FIGURE 1.4 – Le cadre collaboratif d'un algorithme hybride, représentant des structures à plusieurs étages, séquentielles et parallèles .

### 1. Multi-étapes :

Il y a deux étapes impliquées dans ce cas, le premier algorithme agit comme l'optimiseur global tandis que le second algorithme effectue la recherche locale. Cette catégorie peut bien s'inscrire dans le paradigme de la stratégie de l'aigle décrite dans [Yang et al., 2013a]. Le premier algorithme est capable d'explorer l'espace de recherche globalement pour localiser la zone de convergence prometteuse. Puis le deuxième algorithme effectuera des recherches locales intensives telles que la méthode simplex [Yang et al., 2013b]. Un problème difficile dans une telle implémentation est de savoir quand passer au deuxième algorithme. Des mesures telles que la diversité devraient être incorporées pour faciliter les critères de changement. Les travaux de recherche [Ting et al., 2005b;a] utilisaient l'algorithme génétique comme optimiseur global (premier algorithme), avec l'optimiseur de l'essaim de particules (PSO) pour la recherche locale (deuxième algorithme).

### 2. Séquentiel :

Dans cette structure, les deux algorithmes sont exécutés en alternance jusqu'à ce que l'un des critères de convergence soit satisfait. Par souci de simplicité, les deux algorithmes seront exécutés pour un nombre similaire d'itérations avant de passer à l'algorithme suivant.

### 3. Parallèle :

Dans cette structure, Deux algorithmes sont exécutés simultanément, manipulant sur la même population. L'un des algorithmes peut être exécuté sur un pourcentage prédéfini d'un autre algorithme.

#### 1.7.2 Hybrides intégratifs :

Dans cet aspect, un algorithme est considéré comme un subordonné, intégré dans un métaheuristique maître. Pour cette catégorie, le poids contributif de l'algorithme secondaire est d'environ 10 à 20%. Cela implique l'incorporation d'un opérateur manipulateur d'un algorithme secondaire à un algorithme principal. Par exemple, de nombreux algorithmes ont utilisé l'opérateur de mutation de l'algorithme génétique et

l'algorithme d'essaim particule, aboutissant à ce que l'on appelle le PSO génétique ou PSO muté. D'autres recherches peuvent incorporer les techniques de gradient dans l'algorithme principal [Zhang et Xie, 2003, Xie et al., 2002, Varnamkhasti, 2013]. La figure 1.5 illustre les deux approches possibles :

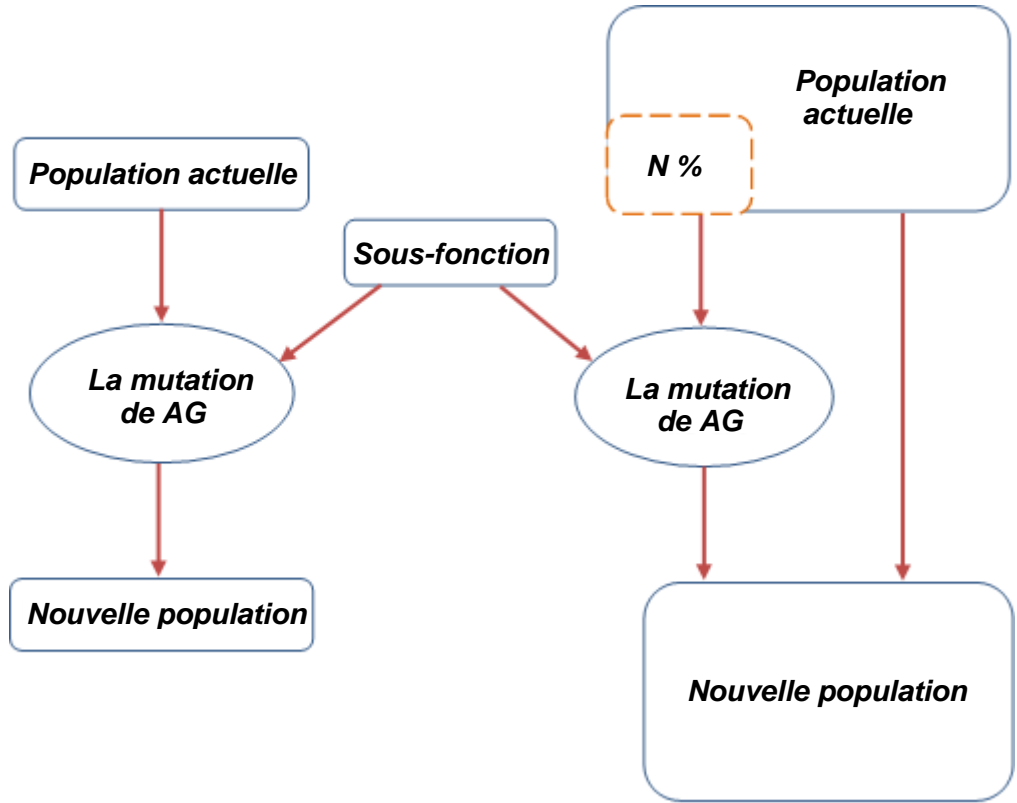


FIGURE 1.5 – Structure intégrative d'un algorithme hybride, avec manipulations complètes et partielles.

**1. Manipulation complète :**

La population entière est manipulée à chaque itération. Une telle opération peut être intégrée en ligne avec le code source existant, généralement en tant que sous-programme/sous-fonction.

**2. Manipulation partielle :**

Dans cette manipulation, seulement une partie de la population entière est accélérée à l'aide de méthodes de recherche locales telles que les méthodes de gradient. Choisir la bonne portion et le bon candidat à accélérer pose un grand défi pour assurer le succès de cette structure hybride.

**1.8 LES CHALLENGES DES ALGORITHMES HYBRIDES :**

Bien que les algorithmes hybrides offrent fondamentalement un grand avantage d'augmenter la diversité dans une population et donc d'amélio-

rer la capacité de recherche de l'algorithme hybride développé, certains inconvénients existent, Ces inconvénients sont décrits comme suit :

### 1.8.1 Convention de dénomination

L'inclusion d'un algorithme dans un autre entraîne généralement un problème de dénomination. Certains chercheurs adoptent des noms très différents pour leurs algorithmes hybrides comme exemple, l'algorithme GA-API [Ciornei et Kyriakides, 2012] est un acronyme pour l'hybridation de " Hybrid Ant Colony-Genetic Algorithm ", ce qui est un peu déroutant pour d'autres chercheurs. Par ailleurs, un nom hybride tel que HPSO-BFGS [Li et al., 2011] semble être une abréviation fastidieuse, plus difficile à lire. En comparaison aux deux architectures mentionnées précédemment, le type collaboratif d'algorithmes hybride semble créer des noms plus sophistiqués et significatifs. Par exemple, il peut être intéressant de comparer les noms de GA-PSO hybride (collaboratif) à PSO muté (intégratif), bien que ces deux hybrides combinent les deux algorithmes de GA et PSO.

### 1.8.2 Complexité de l'algorithme hybride :

En termes d'architecture d'algorithmes, le processus d'hybridation crée généralement des composants supplémentaires dans l'architecture globale de l'algorithme hybride. Cela augmente la complexité de l'algorithme hybride. En raison d'une structure plus complexe, les algorithmes hybrides ont une certaine résistance à être acceptés par les chercheurs. Dans la littérature, deux algorithmes hybrides très populaires sont l'algorithme hybride taguchi-génétique ( Hybrid Taguchi-Genetic Algorithm ) [Tsai et al., 2004] et l'algorithme génétique hybride ( Hybrid Genetic Algorithm ) [Oh et al., 2004], qui ont 434 et 723 citations, respectivement, alors que ces deux algorithmes hybrides sont publiés en 2004. D'après les citations. Il est intéressant de noter que les deux algorithmes relèvent du type intégratif d'algorithme hybride qui possèdent une taxonomie/architecture plus simple.

### 1.8.3 Vitesse de calcul :

Dans de nombreux travaux de recherches, les algorithmes hybrides semblent améliorer les résultats en matière de vitesse de convergence globale et de précision. Cependant, ces résultats de convergence sont souvent évalués par rapport au nombre d'itérations. Cela signifie simplement que la convergence plus rapide ne signifie pas le taux de convergence réel, car l'algorithme hybride utilise un plus grand nombre d'itérations (internes ou implicites). Comme exemple, pour un algorithme hybride collaboratif (du type séquentiel) tel que l'algorithme hybride nommé : GA-PSO, un cycle ou une itération comprend les deux algorithmes GA et PSO. Pour une comparaison équitable, cela doit être considéré comme deux cycles au lieu d'un dans le schéma de convergence. Pour éviter ce problème, le temps d'exécution finale est considéré comme un facteur important et il doit être utilisé comme métrique lors de la comparaison d'un algorithme

hybride avec des algorithmes non hybrides. En outre, en raison d'une architecture plus compliquée dans les algorithmes hybrides, la surcharge se pose parallèlement à sa complexité, dans plusieurs cas est souvent inévitable. Cela affecte les performances globales et tronque ainsi sa robustesse. Pour une comparaison équitable alors le temps de consommation par les frais généraux doit être pris en compte. De nouveau, cela est possible en enregistrant le nombre réel d'itérations prises pour atteindre une cible prédéfinie, bien que les complexités temporelles doivent également être comparées. Il existe d'autres problèmes concernant les algorithmes hybrides. Par exemple, la plupart des algorithmes hybrides augmenteront le nombre de paramètres dans les algorithmes, rendant ainsi plus difficile le réglage de leurs paramètres. De plus, l'analyse est une tâche difficile pour un schéma compliqué d'un algorithme hybride et ne permet donc pas de comprendre les raisons pour lesquelles ces algorithmes hybrides fonctionnent. En outre, les algorithmes hybrides sont légèrement plus difficiles à mettre en œuvre et donc plus sujettes aux erreurs. Ainsi, il faut être prudent lors de l'interprétation des résultats d'algorithmes hybrides.

## 1.9 CONCLUSION

Ce chapitre est consacré pour définir les problèmes d'optimisation ainsi ses classifications, en suite dans la première partie, nous avons décrit les principes des métaheuristiques qui sont un outil indispensable pour résoudre les problèmes d'optimisation complexes et réels. Ils sont particulièrement intéressants pour les problèmes de prise de décision sensibles au temps car, même pour des structures de problèmes très complexes et des instances de problèmes de très grandes tailles, les algorithmes métaheuristiques sont capables de trouver des solutions quasi optimales dans un laps de temps raisonnable. Dans la deuxième partie de ce chapitre, nous avons passé en revue un large éventail d'algorithmes hybrides et étudié les motivations de leurs développements. Nous avons également catégorisé ces algorithmes, basés sur des techniques d'hybridation. De plus, certains inconvénients ont été discutés concernant l'hybridation. Des exemples récents d'algorithmes hybrides de la littérature ont été présentés. Enfin, certaines suggestions ont été recommandées qui peuvent être utiles pour les futurs développements d'algorithmes hybrides.

# LES RÉSEAUX DE CAPTEURS SANS FILS (RCSFs)

# 2

## SOMMAIRE

2.1	INTRODUCTION	27
2.2	HISTORIQUE DES RÉSEAUX DE CAPTEURS SANS FILS	28
2.3	RÉSEAU DE CAPTEURS SANS FIL	30
2.4	PRÉSENTATION D'UN CAPTEUR SANS FIL :	31
2.4.1	Nœud de capteurs sans fil :	31
2.4.2	Type du nœud de capteurs :	32
2.5	DOMAINE D'APPLICATION :	34
2.5.1	Applications militaires	35
2.5.2	Applications médicales	36
2.5.3	Applications en cyber-sécurité	38
2.5.4	Applications environnementales	39
2.5.5	Surveillance d'agriculture	40
2.5.6	Applications aux réseaux véhiculaires	41
2.5.7	Applications aux maisons intelligentes	42
2.6	LES EXIGENCES DU RCSF	43
2.6.1	L'efficacité énergétique	43
2.6.2	L'évolution	44
2.6.3	La réactivité	44
2.6.4	La fiabilité	44
2.6.5	La mobilité	44
2.7	LES CARACTÉRISTIQUES DU RCSF	45
2.7.1	Coût de production	45
2.7.2	Absence d'infrastructure	45
2.7.3	Evolution	45
2.7.4	Interférences	45
2.7.5	Topologie du réseau	46
2.7.6	La sécurité physique limitée	46
2.7.7	la bande passante limitée	46
2.7.8	Le stockage et la récupération	46
2.7.9	La durée de vie	46
2.8	LES PROBLÈMES D'OPTIMISATION DANS LES RCSFs	46

2.8.1	Conception et déploiement . . . . .	47
2.8.2	Agrégation de données . . . . .	48
2.8.3	Protocole de routage . . . . .	49
2.8.4	Clustering . . . . .	50
2.8.5	Qualité de service . . . . .	52
2.8.6	Localisation . . . . .	52
2.9	CONCLUSION . . . . .	54

## 2.1 INTRODUCTION

Le développement de microcontrôleurs, de technologies de communication, de systèmes micro-électromécaniques (MEMS : Micro Electro Mechanical Systems ) et de nanotechnologies a permis la recherche et le développement de nouveaux systèmes de détection et de communication appelée les réseaux de capteurs sans fils (RCSF). Ces réseaux sont caractérisés comme ad-hoc (aucune configuration ou infrastructure d'appui précédent n'est requise), ils utilisent de nouveaux protocoles de communication, surveillent en coopération les phénomènes cibles et envoient les données enregistrées vers une station centrale de traitement appelée station de base (SB). Comme signifie le mot " sans fil " , ces réseaux de capteurs communiquent à l'aide des canaux de communication sans fil ce qui permet un déploiement, un contrôle, une maintenance et manipulation de capteurs possibles. Les capteurs sans fils dans les systèmes en réseau sont généralement appelés nœuds, car ils sont construits avec plus de composants que de simples capteurs. Du point de vue matériel, les nœuds de capteurs sont des ordinateurs intégrés de petite taille associée à une variété de capteurs choisis par l'utilisateur en fonction de l'application visée (détecter les paramètres de température, lumière et d'humidité, etc.). Les nœuds de capteurs ont généralement des microprocesseurs ou micro-contrôleurs intégrés sous forme d'ensemble de composants : batterie, mémoire, radio, ports de communication, circuits d'interface et enfin de capteurs pour des applications spécifiques. Ce sont des appareils intégrés complexes qui combinent les techniques informatiques, de communication et de technologies des capteurs. Étant donné un réseau de petits appareils intégré, les RCSFs sont totalement différents des réseaux informatiques générales telles qu'Internet ou Ethernet. Les réseaux de capteurs sans fils (RCSFs) ne possèdent pas de topologies caractéristiques comme les réseaux locaux (Local Area Network) tels que bus, anneau ou étoile. Il s'agit principalement de réseaux ad-hoc déployés de façon aléatoire dans les champs de captage et basé principalement sur la norme sous-jacente IEEE 802.15.4, ces normes sont largement adoptées pour les appareils embarqués. Ils sont spécifiques à l'application de la communication en réseau et parfois conçus spécifiquement pour accueillir des applications ciblées. Les RCSFs sont limité par de nombreuses contraintes : généralement absent dans les réseaux généraux, à titre d'exemple : la disponibilité limitée d'énergie et de bande passante, le petit facteur de forme, la quantité des nœuds déployés sur des zones grandes ouvertes, etc. La mise en réseau et la communication des RCSFs doivent être ajustées de façon créative pour prendre en charge des applications spécifiques, de plus, une nouvelle optimisation inter-couches et des changements dans la communication des protocoles ont été développés pour répondre aux exigences spécifiques des réseaux de capteurs. Ce type de réseaux est sujet à de nombreuses contraintes, difficultés environnementales et technologiques en fonction des besoins des entreprises. Les RCSFs ont évolué et connu par de nombreuses caractéristiques qui les distinguent des autres types de réseaux. Ils ont une grande capacité de fonctionner sans surveillance avec/ou sans supervision. Les principaux composants du RCSF qui sont les nœuds de capteurs, sont peu coûteux et généralement à usage unique et ils prennent en charge les

topologies dynamiques qui peuvent surmonter les pannes de nœuds ou de capteurs, les pertes de liaisons de communication ou pendant le mouvement des nœuds. Chaque nœud est constitué d'un module de détection, d'un module de réception et d'un module de transmission. Par ailleurs, les nœuds peuvent également fonctionner dans des environnements difficiles et dangereux avec un opérateur humain à une distance de sécurité. En raison de leur petite taille et la non-existence des câbles, les RCSFs n'ont aucune influence perturbatrice sur l'environnement ou les processus industriels. Par rapport aux capteurs individuels affectés à la mesure et à l'observation de phénomènes spécifiques, les RCSFs ont la capacité et la flexibilité de travailler de manière coopérative, d'échanger les données et de les traiter individuellement dans le réseau. Finalement, dans ce chapitre nous donnons d'abord un aperçu sur les réseaux de capteurs qui sont un super ensemble des réseaux de capteurs sans fils, définition, ses composants en détail et ainsi ses caractéristiques puis on décrit en détail le domaine d'application de ce type de réseau, les challenges rencontrés dans les différents niveaux et on terminera par une conclusion.

## 2.2 HISTORIQUE DES RÉSEAUX DE CAPTEURS SANS FILS

Le développement des RCSFs a été inspiré par des applications militaires, une surveillance notable dans les zones de conflit. La recherche sur les RCSFs remonte au début des années 1980, lorsque l'agence des projets de Défense avancées (Defense Advanced Research Projects Agency (DARPA)) a lancé le programme de réseaux de capteurs distribués (DSN :Distributed Sensor Network) pour l'armée américaine. À cette époque, le réseau d'agences de projets de recherche avancée (ARPANET) fonctionnait depuis plusieurs années, avec environ 200 hôtes dans les universités et les instituts de recherche [Chee-Yee Chong et Kumar, 2003]. Les DSNs étaient supposés avoir de nombreux nœuds de détection à faible coût réparti dans l'espace, collaborant entre eux mais fonctionnant de manière autonome, les informations étant acheminées vers le nœud qui peut le mieux utiliser les informations. Même si les premiers chercheurs sur les réseaux de capteurs avaient en tête la vision d'un DSN, la technique n'était pas tout à fait prête. Plus précisément, les capteurs étaient plutôt grands (c'est-à-dire la taille d'une boîte à chaussures et plus gros), et le nombre d'applications potentielles était donc limité. De plus, les premiers DSNs n'étaient pas étroitement associés à la connectivité sans fil. Les récents progrès de l'informatique, de la communication et de la technique micro-électromécanique ont entraîné un changement significatif dans la recherche sur le RCSF et l'ont rapprochée de la vision originale. La nouvelle vague de recherche sur les RCSFs a commencé vers 1998 et attire de plus en plus l'attention et l'implication internationale. Cette vague de recherche a mis l'accent sur la technique de mise en réseau et le traitement de l'information en réseau adapté aux environnements ad-hoc hautement dynamiques et aux nœuds de capteurs à ressources limitées. De plus, les nœuds de capteurs ont été beaucoup plus petits (c'est-à-dire de celui d'un paquet de cartes à la poussière particules) et beaucoup moins cher dans le prix et donc de nombreuses nouvelles applications civiles des réseaux de capteurs tels que la surveillance de l'environnement, le réseau

des capteurs de véhicules et les réseaux de capteurs de corps ont vu le jour. Encore une fois, DARPA a joué un rôle de pionnier dans la nouvelle vague de recherche sur les réseaux de capteurs en lançant un programme de recherche d'initiative appelé SensIT [Kumar, 2001] qui a fourni aux réseaux de capteurs actuels de nouvelles capacités telles que la mise en réseau ad hoc, l'interrogation et la tâche dynamiques, la reprogrammation, en fait, les RCSFs sont considérés comme l'une des techniques les plus importantes du 21<sup>me</sup> siècle [More et Raisinghani, 2017]. À titre d'exemple, la Chine a inclus les RCSFs dans ses programmes nationaux de recherche stratégique [Ni, 2008]. En conséquence, la commercialisation des RCSFs s'accélère et de nombreuses sociétés de nouvelles technologies émergent telles que Crossbow Technology (connectant le monde physique au monde numérique) et la société Dust Networks. Aujourd'hui, l'automatisation industrielle est l'un des domaines les plus importants des applications des RCSFs. Selon la société Freedonia Group, la part du marché mondiale des capteurs à usage industriel est de 11 milliards de dollars, tandis que le coût d'installation (principalement les coûts de câblage) et d'utilisation atteint plus de 100 milliards de dollars. Ce coût élevé est le principal problème entravant le développement de la technique de communication industrielle. La technique des RCSFs permettant une détection omniprésente sur l'ensemble du processus industriel et peut sécuriser les paramètres importants qui ne sont pas disponibles par la surveillance en ligne pour les raisons de coût indiquées ci-dessus. Ces paramètres sont des bases importantes pour la mise en œuvre d'un contrôle optimal afin d'atteindre l'objectif d'amélioration de la qualité des produits et de réduction de la consommation d'énergie. Selon ON World [Rodenas-Herraiz et al., 2013], les dispositifs sans fils à installer dans les domaines industriels augmenteront de 53 % entre la période de 2011 et 2016, alors qu'il y aura 24 millions de capteurs et d'actionneurs sans fils ou points de détection, déployés dans le monde entier. Parmi ceux-ci, 39% seront utilisés pour de nouvelles applications ne sont pas possibles qu'avec la mise en réseau de capteurs sans fil. D'ici 2022, le nombre d'appareils représentera 65 % de l'ensemble des points de détection des équipements de mesure et de contrôle industriels et 77 % d'ici 2016. Sur le marché actuel, les trois quarts des revenus industriels du RCSF proviennent de l'industrie des procédés, l'industrie pétrolière et électrique étant celle qui connaît la croissance la plus rapide. Par exemple, le plus gros groupe pétrolier de Chine (**Petro China**) mène des projets Internet des objets dans ses champs pétroliers, dans le but de reconstruire 200 000 puits de pétrole.

La technologie du RCSF a été appliquée dans les conversions numériques des puits de pétrole et fera usage de la surveillance en ligne pour mesurer la production du puits de pétrole et assurer la sécurité de la production. Dans le secteur de l'énergie qui subit actuellement la mise à niveau du réseau électrique, la technologie du RCSF joue également un rôle important dans la surveillance de la sécurité des équipements de transmission et de transformation de l'énergie et la reconstruction de milliards de compteurs intelligents.

### 2.3 RÉSEAU DE CAPTEURS SANS FIL

Au cours des dernières années, les applications des réseaux de capteurs sans fils (RCSFs) ont connu une attention très remarquable [M, 2017] [Kochhar et Kumar, 2013] [Azzedine Boukerche, 2018] [N et al., 2017] [HTT et al., 2018] [Roy et al., 2016] et l'utilisation de ce type du réseau est devenu une partie intégrante de nos vies. Les RCSFs sont composés de systèmes intégrés individuels capables de :

1. Interagir avec leur environnement grâce à divers capteurs.
2. Traitement local des informations de manière autonome.
3. Communiquer ses informations sans fil avec leurs voisins.

Le RCSF se compose d'un ensemble de nœuds de capteurs et d'une Station de Base (SB) qui se sont répartis dans un environnement de détection (captage) pour réaliser un objectif bien déterminé (spécifique). SB est un système de coordination principale ayant des capacités de haute énergie et du traitement élevé [Barnawi et al., 2019]. [S et al., 2018]. Le nœud du capteur se compose de plusieurs éléments tels que l'antenne interne et externe, le microcontrôleur, la batterie. Etc. Tous les capteurs connectés sont autonomes et ils sont directement ou indirectement connectés à la SB. Les nœuds de capteurs sont utilisés pour détecter certaines informations de l'état physique ou de l'environnement. Ces informations comprennent la lumière, la chaleur, la pression, etc. **La figure 2.1** illustre les composants d'un RCFS . Sur cette figure-là, différents nœuds de capteurs sont connectés avec une portée radio à une autre portée radio. Chaque nœud de capteurs réside dans une portée radio.

**la figure 2.1** contient des nœuds de capteurs de différentes couleurs. La collection de nœuds de capteur est connectée avec la station de base et

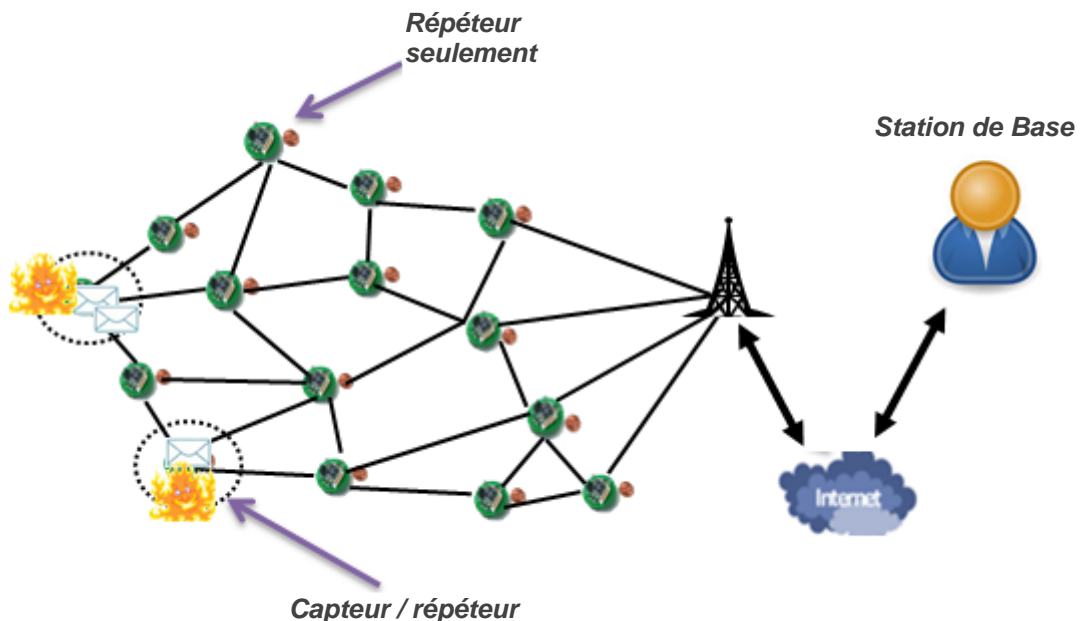


FIGURE 2.1 – Réseau de capteurs sans fil.

la station de base est en outre connectée avec l'utilisateur final. Cet utilisateur final est connecté à la connexion Internet. La station de base (SB)

est également appelé parfois le nœud récepteur, qui est chargé de collecter les données et les informations de tous les nœuds capteurs. Noter que ce nœud est capable de traiter les données détectées et de transmettre des informations résumées (demandé) à l'utilisateur final. L'utilisateur final peut être une personne ou un logiciel ou un matériel conçu pour un usage spécifique. En fait, l'utilisateur final indique directement ou indirectement " objectif final ". Cet objectif indique l'exportation des informations utiles au client qui remplit le véritable objectif de conception du RCSF.

## 2.4 PRÉSENTATION D'UN CAPTEUR SANS FIL :

Les capteurs sont des dispositifs de taille extrêmement réduite avec des ressources très limitées, autonomes, capables de traiter des informations et de les transmettre, via les ondes radio, à une autre entité (capteurs, unité de traitements) sur une distance limitée à quelques mètres. **La figure 2.2**, illustre le schéma interne d'un capteur sans fil. Les réseaux de capteurs utilisent un très grand nombre de ces capteurs, pour former un réseau sans infrastructure. Un capteur analyse son environnement, et transmet les données collectées aux capteurs appartenant à sa zone de couverture. Chaque capteur relayant l'information sur sa propre zone de couverture, donc le réseau se trouve entièrement couvert.

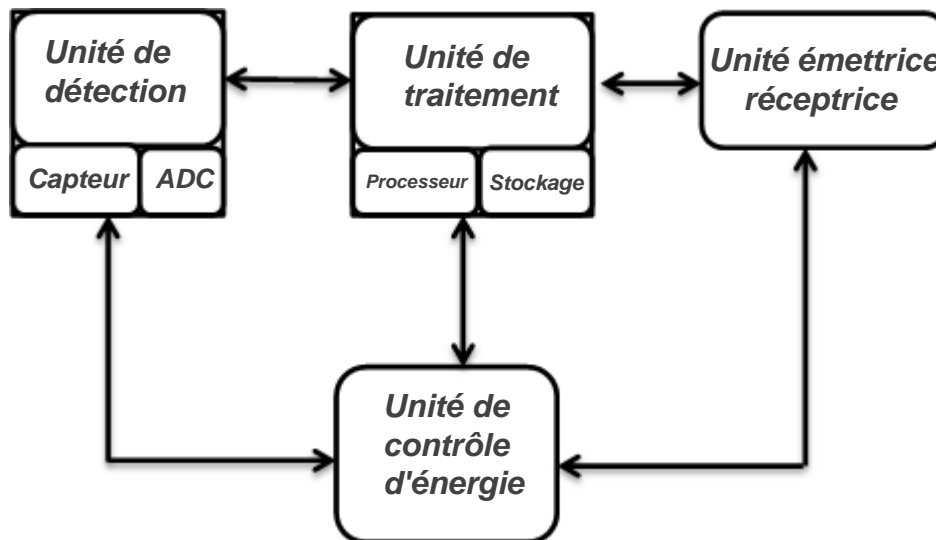


FIGURE 2.2 – Les composants d'un capteur sans fil.

### 2.4.1 Nœud de capteurs sans fil :

Un nœud capteur se compose de quatre unités de base : l'unité de captage, l'unité de traitement, l'unité de transmission et l'unité de contrôle d'énergie. Il peut contenir également, selon son domaine d'applications, des modules supplémentaires tels qu'un système de localisation (GPS), ou bien un système générateur d'énergie (cellule solaire). On peut même trouver des micro-capteurs, un peu plus volumineux, dotés d'un système mobilisateur chargé de déplacer le micro-capteur en cas de nécessité :

#### **Unité de détection :**

Sont généralement composées de deux sous-unités, des capteurs et des convertisseurs analogiques-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.

#### **Unité de traitement :**

L'unité de traitement est généralement associée à une petite unité de stockage, c'est une unité chargée d'exécuter les protocoles de communications qui permettent de collaborer le nœud de capteur avec les autres nœuds pour effectuer les tâches de détection affectées.

#### **Unité émettrice-réceptrice :**

Cette unité est responsable d'effectuer toutes les émissions et réceptions des données sur un médium sans fil. Elle peut être du type optique (comme dans les nœuds Smart Dust), ou de type radiofréquence. Les communications du type optique sont robustes vis-à-vis des interférences électriques. Néanmoins, elles présentent l'inconvénient d'exiger une ligne de vue permanente entre les entités communicantes. Par conséquent, elles ne peuvent pas établir de liaisons à travers des obstacles.

#### **Unité de contrôle d'énergie**

Un capteur est muni d'une ressource énergétique pour alimenter tous ses composants. Cependant, en conséquence de sa taille réduite, la ressource énergétique dont il dispose est limitée et généralement irremplaçable. Cette unité peut aussi gérer des systèmes de rechargement d'énergie à partir de l'environnement observé tel que les cellules solaires, afin de prolonger la durée de vie du réseau. Des composants supplémentaires dépendant de l'application telle qu'un système de localisation, un générateur électrique et un mobilisateur. La plupart des techniques de routage du réseau de capteurs et des tâches de détection nécessitent la connaissance de l'emplacement avec une grande précision, il est donc courant qu'un nœud de capteur dispose d'un système de localisation. Un mobilisateur peut parfois être nécessaire pour déplacer les nœuds de capteurs lorsqu'il est nécessaire pour effectuer les tâches assignées.

Quelques types de capteurs sans fils sont présentés dans **la figure 2.3** .

#### **2.4.2 Type du nœud de capteurs :**

Le capteur est utilisé dans une large gamme des contraintes qui correspondent aux formes de salubrité, tels que les mouvements, les signaux électriques, l'énergie thermique et contraignante. Les nœuds de capteurs sont principalement deux types :

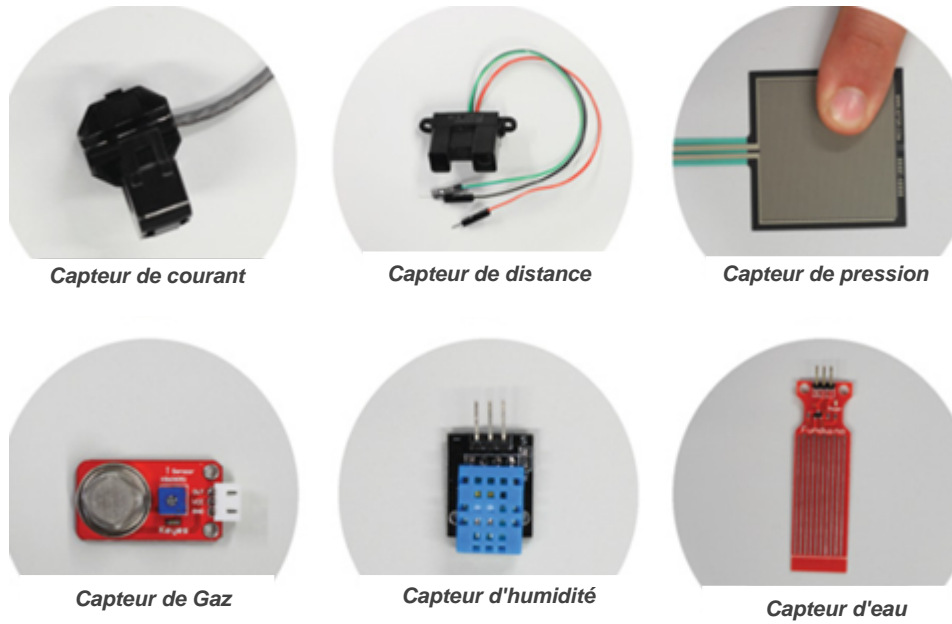


FIGURE 2.3 – Types des capteurs sans fil.

### Les capteurs passifs :

Les capteurs passifs détectent les données par sondage actif, ce qui signifie que leur énergie n'est utilisée que pour amplifier les signaux analogiques. Les capteurs passifs ne nécessitent pas de ressources externes, car leur énergie est utilisée uniquement pour générer des signaux analogiques. Ces capteurs peuvent modifier ses propriétés physiques (comme la capacité) et générer des signaux électriques, aussi, ces capteurs passifs sont divisés en deux types de capteurs passifs omnidirectionnels et de capteurs passifs à faisceau étroit. Pour les nœuds de capteurs omnidirectionnels passifs, une zone géographique particulière est attribuée à chaque nœud de capteurs de sorte que le nœud de capteurs de la zone se déplace et collecte des informations fiables et les transmette à la station de base pour un traitement ultérieur. Les travaux de recherche actuelle supposent que les nœuds de capteurs sont omnidirectionnels. Cependant, pour les nœuds de capteurs à faisceau étroit passifs, une direction ou une notation prédéterminée est donnée à chaque nœud de capteurs de la même manière que les caméras qui sont capables de prendre des photos dans une seule direction. Les nœuds de capteurs peuvent déplacer et collecter des données et aussi les transmette à la station de base dans une seule direction.

### Les capteurs actifs :

Les nœuds de capteurs actifs détectent les données en manipulant l'environnement, ce qui signifie qu'ils ont besoin d'une énergie continue pour générer des signaux analogiques. Les nœuds de capteurs actifs sondent dynamiquement l'environnement voisin et utilisent une alimentation électrique externe, qui est nécessaire pour surveiller et fonctionner. Les capteurs numériques génèrent des indications distinctes qui sont en-

suite converties en caractérisation numérique en fonction des paramètres mesurés. Une sortie de valeurs discrètes (groupe d'octets) uniques est mesurée représente la quantité obtenue. Les capteurs analogiques génèrent des signaux continus (tels que la température, la pression etc.) qui sont mesurés sous forme de signaux analogiques continus. Chaque nœud de capteur est également appelé une mote, mais toutes les particules ne sont pas toujours appelées nœuds de capteur. Motes reçoit et transmet les données et utilise certaines techniques de traitement à l'intérieur du nœud de capteur, comme la compression, le cryptage, etc. Ainsi, les nœuds de capteur fonctionnent comme des motes, mais les motes ne fonctionnent pas toujours comme des nœuds de capteurs [1–3]

Les capteurs reçoivent et transmettant les données en utilisent certaines techniques de traitement spécifique, comme la compression, le chiffrement, etc. Ainsi, le nœud de capteurs fonctionne comme des capteurs [Islam et al., 2015, Holler et al., 2014, Zhibo, 2013]. La dernière décennie a connu un progrès rapide dans les techniques de détection multiple :

- Systèmes micro électromécanique (MEMS) - gyroscopes, accéléromètres, magnétomètres, capteurs de pression, capteurs à effet pyro-électrique, capteurs acoustiques [Zhu et al., 2019].
- Capteurs CMOS - température, humidité, proximité capacitive, composition chimique [C.Eder et al., 2014].
- Capteur LED - détection de lumière ambiante, détection de proximité, composition chimique [Nair et Prakash, 2015].

## 2.5 DOMAINE D'APPLICATION :

Dans le système des réseaux de capteurs sans fils, de nombreux nœuds de capteurs radio collaborent pour permettre la communication en absence d'infrastructure fixe. Avec la flexibilité et l'évolution, les réseaux de capteurs sans fils ont un grand potentiel pour une variété d'applications, L'objectif fondamental des nœuds de capteurs dans le RCSF est de surveiller les conditions environnementales ou physiques telles que le son, la pression, la température, etc. Ensuite, ils collectent les données et les envoient à la station de base. Fondamentalement, le RCSF est conçu en premier lieu pour des applications militaires où il est rapidement utilisé pour la surveillance du champ de bataille. En raison de plusieurs avantages, RCSF est aujourd'hui utilisé à plusieurs objectifs. L'une des installations modernes en pleine croissance du RCSF est le réseau sans fil basé sur l'Internet des objets (IoT : Internet of Things) dans le système de santé [Skiadopoulos et al., 2019]. [Riaz et al., 2017] L'internet des objets a ouvert la voie au système traditionnel comme à un hôpital de visite. L'internet des objets permet des fonctionnalités telles que la communication, la détection, le traitement avec des paramètres biomédicaux et physiques [A et GP, 2019]. [Karati et al., 2018b]. L'informatique en nuage offre également certains avantages au système de santé, car il possède une grande infrastructure de traitement et de stockage. Il aide à traiter les données et informations en mode hors-ligne ainsi qu'en ligne par des flux de capteurs corporels [Panda et Jana, 2019]. [Panda et Jana, 2018]. [Panda et al., 2018]. [Karati et al., 2018a]. autre les applications militaires et l'internet

des objets, il existe plusieurs applications du RCSF, on a les cités comme suit :

### 2.5.1 Applications militaires

Les réseaux de capteurs sans fils sont largement utilisés dans les applications militaires, la mise en œuvre du réseau de capteurs sans fil a commencé principalement avec l'application militaire avec la formation de DARPA et le suivi de l'ennemi. Étant donné que les nœuds capteurs sont densément déployés, si certains nœuds capteurs sont détruits par des actions antagonistes, cela n'influence pas ou n'affecte pas l'opération militaire. Cela rend les nœuds de capteurs idéaux pour le champ de bataille. En utilisant des capteurs appropriés dans la zone de réseau spécifique, l'identification des forces ennemies, la détection des mouvements, l'analyse et la progression de leurs mouvements peuvent être atteints. Les nœuds capteurs fournissent les services suivants dans le champ de bataille [Ko et al., 2010] :

**Surveillance du champ de bataille** les nœuds capteurs sont déployés dans la région frontalière pour suivre tout intrus qui ont l'intention de franchir les frontières.

Lors de la détection d'un tel intrus,

les nœuds capteurs envoient leurs informations à la station de base la plus proche et les personnes autorisées peuvent prendre les mesures nécessaires pour perturber toute activité irrégulière. De même, les réseaux de capteurs peuvent être utilisés pour suivre une cible particulière, un processus connu sous le nom de suivi de cible. **La figure 2.4** Montre un exemple de suivi de cible à l'aide de réseaux de capteurs.

Les nœuds de capteurs sont déployés dans une région pour suivre différents objets. Donc, les nœuds de capteurs détectent le mouvement des objets et se rapportent à la station de base. Le capteur peut également être activé en fonction du mouvement des objets pour les suivre de manière efficace.

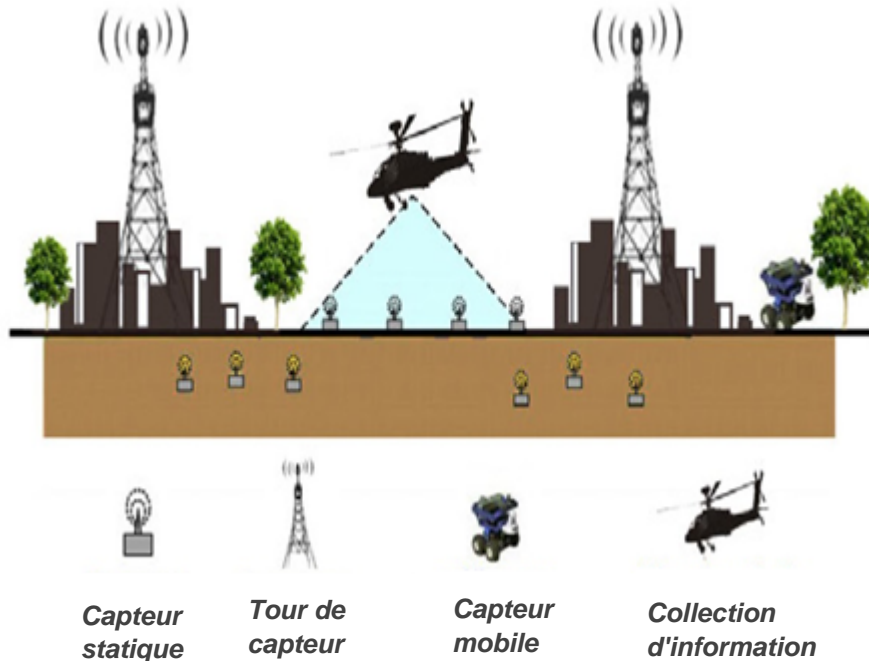


FIGURE 2.4 – Suivi de cible sur le champ de bataille.

**Ciblage** les capteurs sont placés dans des armes et collectent des informations sur les cibles (comme le mouvement, le mouvement, l'angle, la distance, etc.) qui peuvent être envoyées au tireur. Ce capteur est utilisé pour une meilleure estimation de la cible.

**Détection d'intrusion** Les nœuds de capteurs sont utilisés pour identifier l'intrusion dans le réseau et sonner une alarme. Le système de défense évaluera alors un recours approprié pour l'attaque, un système de détection d'intrusion est présenté dans la figure 2.5 . [Butun et al., 2014]. [Pundir et al., 2020]

**Forces de surveillance** les nœuds de capteurs surveillent le mouvement des troupes et des chars et signalent ces informations à la station de base.

### 2.5.2 Applications médicales

Le système de santé est basé sur le RCSF se compose des nœuds de capteurs autonomes qui communiquent entre eux à l'aide d'une communication sans fil. Ces nœuds mesurent les paramètres physiques tels que le mouvement, la température, la pression de la zone de l'intérêt, etc. Le système de soins de santé à distance fournit une assistance pour surveiller l'avancement de la maladie et suivre le patient individuel. Les patients sont examinés par les médecins et ils sont conseillés de suivre certaines précautions pendant une période de temps donnée. Ainsi, le système de

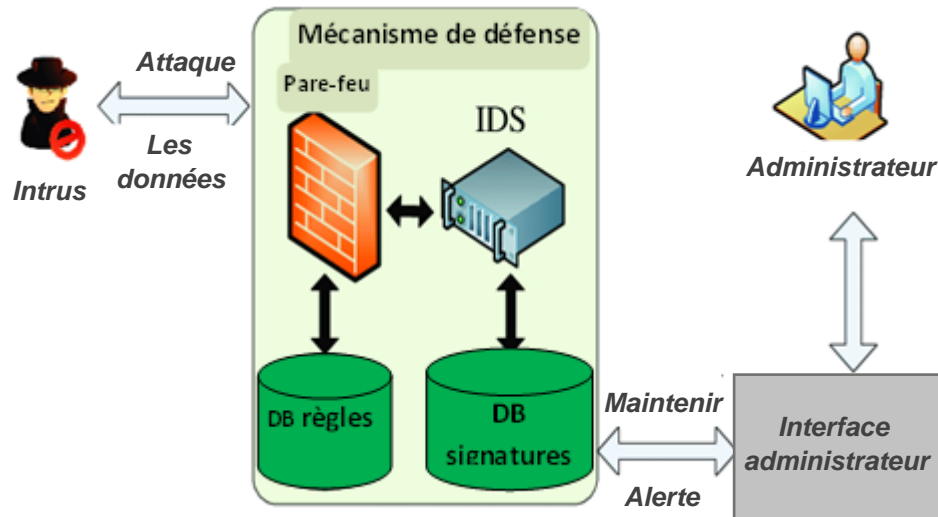


FIGURE 2.5 – Détection d'intrus à la frontière de la région.

surveillance des soins de santé fournit un soutien aux assistants à domicile des patients âgés et handicapés [Sharma et Bhatt, 2018]. Les informations du patient doivent toujours rester secrètes du domaine public, car ces informations peuvent être utilisées de façon inappropriée par d'autres personnes.

Le système de santé basé sur le RCSF peut être largement classés en e-santé, m-santé, soins à distance.

**E-santé** [Sahi et al., 2018] est un système interactif et informatif où les patients et les professionnels de la santé peuvent se réunir et partager des informations à travers le monde. Un système de santé permet aux professionnels de la santé et / ou aux médecins d'analyser des données et de fournir la meilleure solution possible pour les patients.

Les dossiers de santé électronique contiennent tous les d'informations sur les patients, mais le problème est que les informations relatives à la vie privée du patient seront divulguées à la société.

**M-santé** [Kang et Nyang, 2017] Comprends l'utilisation d'un appareil mobile pour la collecte de données sur la santé des patients. Le système m-santé aide à fournir des informations aux patients et aux chercheurs par une surveillance en temps réel grâce à la télémédecine qui traite et aide la prestation directe des soins. Les progrès des techniques du RCSF et leur utilisation dans les applications de santé ont connu un progrès très rapide [R.R. et al., 2016], [Rojas et Barrett, 2017]. Certaines applications telles que la surveillance de la fréquence cardiaque (ECG), la surveillance de la pression artérielle (blood pressure monitoring), etc. ils sont largement utilisés dans notre vie quotidienne. Le réseau corporel sans fil (wireless body area network) est un autre domaine qui a été développé pour répondre à l'avancement de la technique des capteurs [Qi et al., 2017]. Un réseau corporel sans fil est présenté dans la **figure 2.6**, il permet de traiter les informations collectées sur le corps de la personne sur un réseau de communication local et de les transmettre aux services de santé concernées, où les professionnels du service de santé peuvent surveiller à distance le patient.

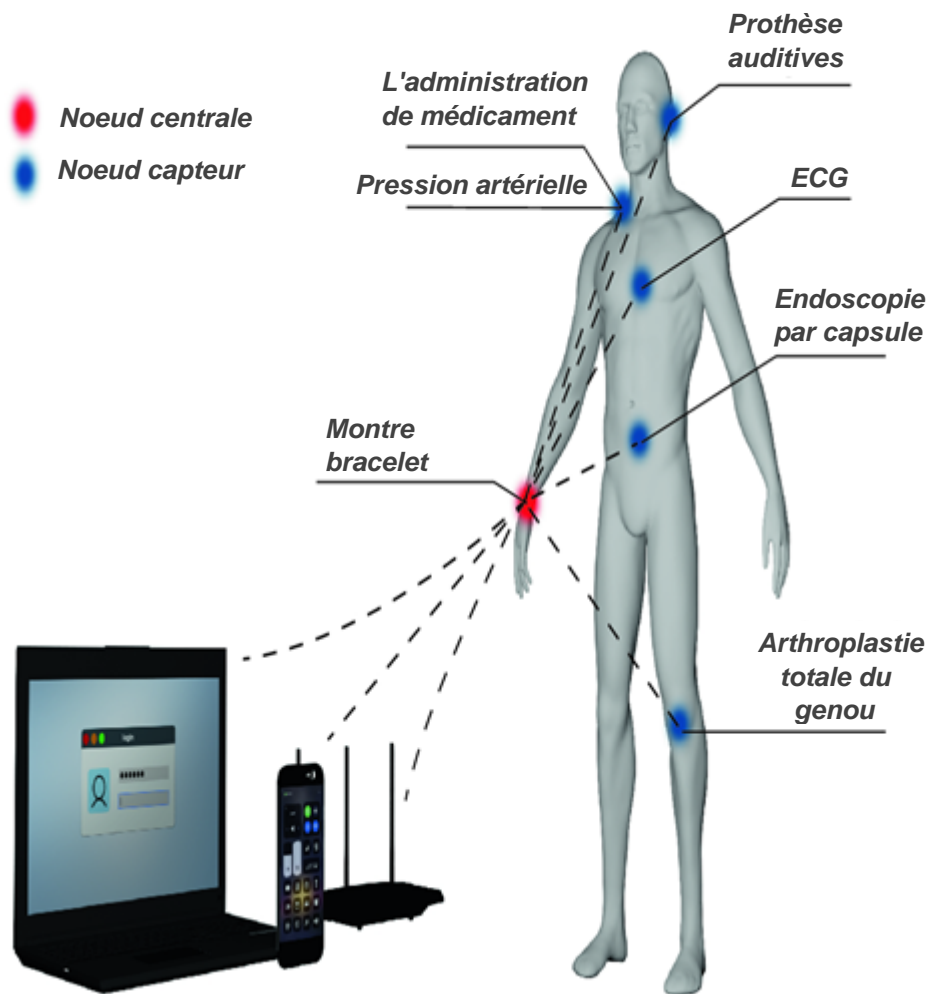


FIGURE 2.6 – Réseau corporel sans fil.

### 2.5.3 Applications en cyber-sécurité

Un système CyberPhysique, Cyber-Physical System (SCP) est un système où des éléments informatiques collaborent pour le contrôle et la commande des entités physiques.

Le déploiement généralisé des Technologies de l'Information (TI) dans divers SCPs tels que les réseaux intelligents et les réseaux informatiques les ont rendus vulnérables à divers types d'attaques de sécurité appelée cybers-attaques. De telles attaques deviennent de plus en plus dangereuses, tentant d'obtenir un accès non autorisé à un service ou à des données, ou tentant de compromettre la confidentialité, la disponibilité, l'intégrité.

Ces dernières années ont entraîné une augmentation considérable du nombre de cybers-attaques, ainsi que l'émergence de divers types de cybercriminels qui développent constamment de nouvelles techniques d'attaque [Mahdavifar et Ghorbani, 2019]. Les SCPs adoptent et déploient largement des techniques, telles que les réseaux de capteurs sans fils pour de nombreux domaines d'application. En particulier, les RCSFs contribuent à assurer la cybersécurité des SCPs, où les capteurs peuvent collecter dyna-

miquement des informations physiques grâce à un processus coopératif qui permet de détecter et d'atténuer les futures cybers-attaques potentielles, (voir la figure 2.7).

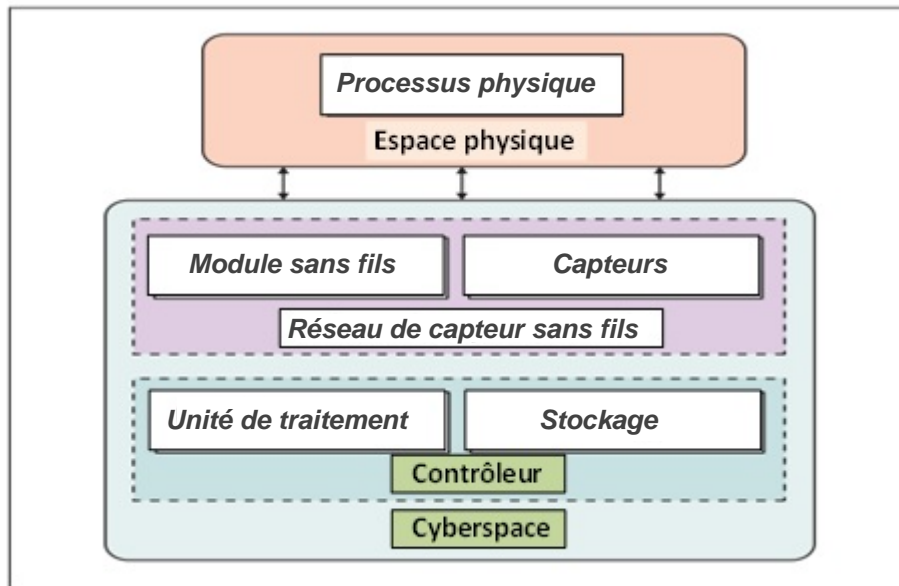


FIGURE 2.7 – Système cyber-physique compatible au RCSF.

Dans l'autre côté, les RCSFs ont été largement utilisées pour soutenir diverses fonctions de surveillance et de sécurité. Par exemple, des réseaux de capteurs ont été déployés pour soutenir des capacités de surveillance telles que la détection de la présence des menaces dans des régions sensibles à la sécurité et hostiles telles les zones militarisées, la protection des frontières, etc. Pour renforcer la surveillance, un modèle de RCSF a été proposé dans [Ghataoura et al., 2011] pour détecter et déterminer la direction du mouvement du personnel et des véhicules importuns (suivre la cible) dans une zone sensible. Par ailleurs, pour de nombreuses fonctions de surveillance, les nœuds de capteurs travaillent en coopération entre eux pour détecter une approche imminente de la menace mobile et aussi sont capables de s'auto-organiser pour fournir une vue claire, pertinente et optimale du réseau de surveillance. Ces informations recueillies aident à améliorer les capacités de prise de décision pour le commandement et le contrôle, le renseignement, la surveillance et la planification de missions tactiques de reconnaissance.

En outre, pour objectif d'améliorer ses capacités de prise de décision, les nœuds de capteurs devraient être en mesure de transmettre des informations sur les menaces à un nœud de passerelle appelé nœud récepteur. Dans un tel l'environnement, les nœuds de capteurs peuvent être compromis par des intrus pour perturber les données détectées et transmises en injectant de faux rapports de données [Yuan et al., 2019].

#### 2.5.4 Applications environnementales

Les nœuds de capteurs jouent un rôle important pour superviser et évaluer les conditions environnementales de grandes zones géogra-

phiques (la température, l'humidité, les précipitations, etc.) , ils aident la surveillance de la pollution, à la détection des incendies de forêt et à la détection des inondations. Le premier RCSF bien connu pour la couverture météorologique était une évaluation automatisée du monde réel en temps réel (ALERTE). Elle peut produire des informations en temps réel concernant les précipitations et les niveaux d'eau pour estimer possibilité d'inondation. Un système de détection automatique d'incendie est présenté dans **la figure 2.8**. La surveillance de l'environnement est largement divisée en deux types, surveillance intérieure et extérieure. La surveillance intérieure est utilisée pour les bâtiments et les bureaux. Cette application implique de détecter l'humidité, la température et la qualité de l'air. Des exemples d'applications de surveillance extérieure sont les éruptions volcaniques, les tremblements de terre, la détection des risques chimiques et les prévisions météorologiques . [Granda et al., 2018].

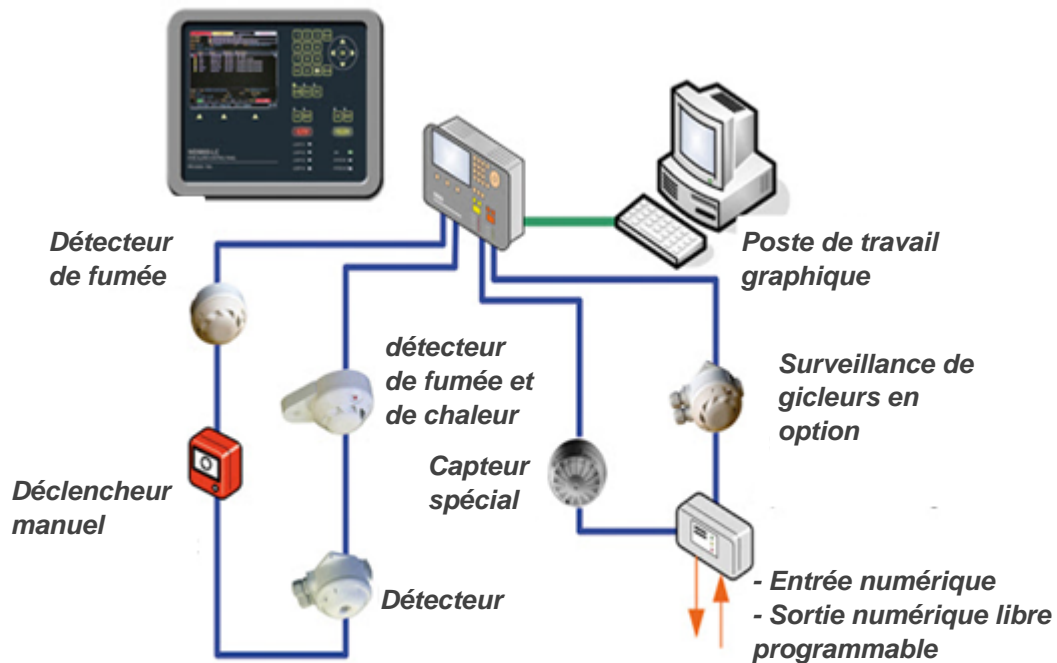


FIGURE 2.8 – Système de détection automatique d'incendie.

### 2.5.5 Surveillance d'agriculture

Récemment, des réseaux de capteurs ont été déployés dans les champs agricoles pour surveiller des paramètres tels que la température du sol et l'humidité. **La figure 2.9** illustre un champ agricole avec le déploiement des nœuds de capteurs. Différents nœuds de capteurs sont déployés sur le terrain et détectent les paramètres correspondants [Rathinam et al., 2019]. Les données détectées sont transmises aux passerelles et éventuellement aux serveurs distribués où les informations sont traitées et calculées pour permettre des décisions optimales. Selon les informations traitées, les utilisateurs distants sont avertis de prendre les mesures nécessaires (par

exemple, fournir de l'eau). Ainsi, un système d'agriculture de précision peut être déployé à l'aide des réseaux de capteurs sans fils.

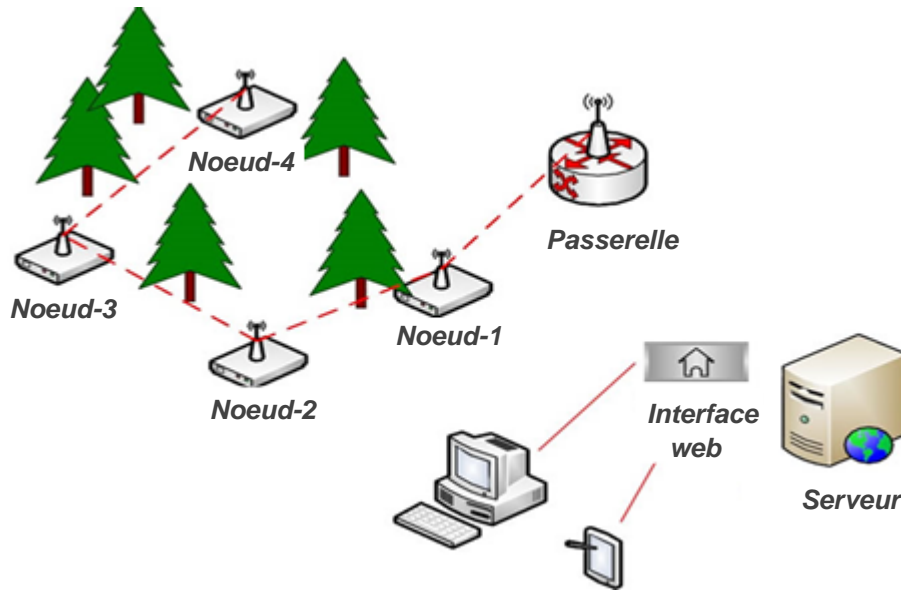


FIGURE 2.9 – Réseau de capteur agricole.

### 2.5.6 Applications aux réseaux véhiculaires

Les RCSFs jouent un rôle clé pour permettre des systèmes de transport intelligent (STI) où les véhicules forment un réseau et échangent des informations. Les véhicules téléchargent leurs informations collectées sur les Unités routières (UR) et téléchargent des informations à partir des RSUs pour obtenir des mises à jour récentes du serveur [Bera et al., 2015]. Les informations collectées sont transférées à la station de base pour le calcul et le traitement. Dans la figure 2.10, un réseau des véhicules est présenté où les véhicules communiquent entre eux et échangent des informations pour assurer une meilleure conduite.

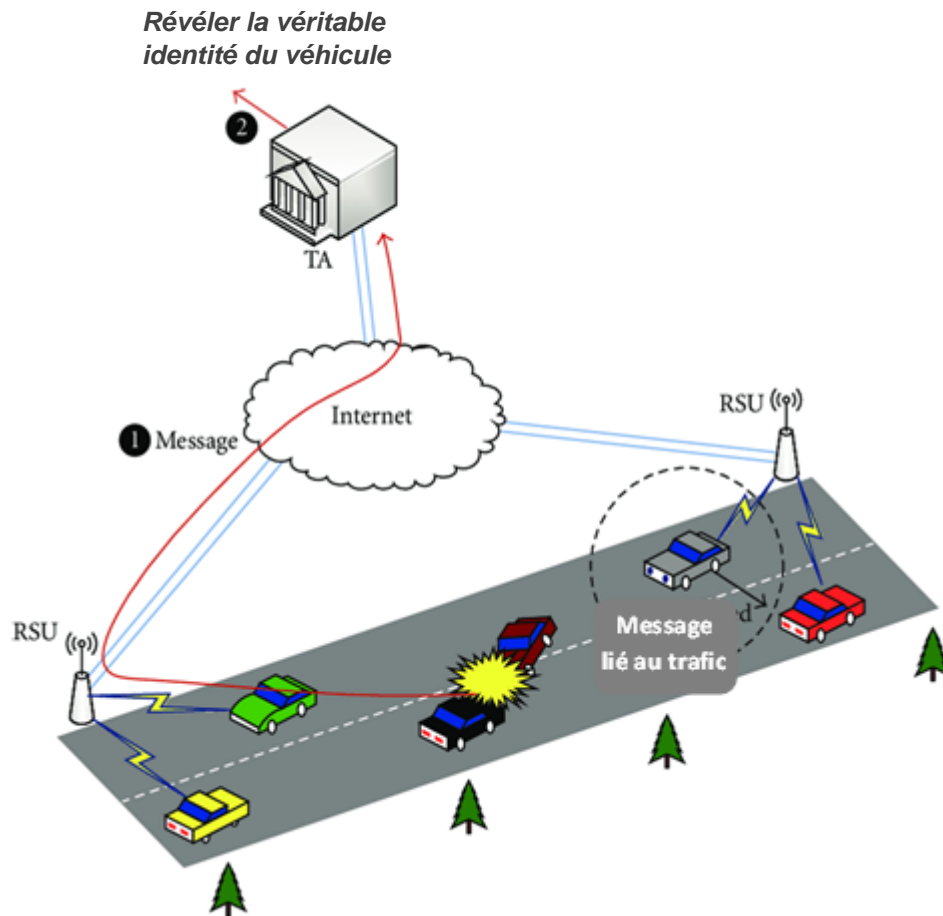


FIGURE 2.10 – Réseau de capteurs sans fil du réseau routier.

### 2.5.7 Applications aux maisons intelligentes

En raison des préoccupations croissantes concernant la crise de l'énergie, les gens recherchent des systèmes automatisés d'approvisionnement en énergie numérisés [Maldonado, 2017]. La gestion à distance des applications domestiques est réalisable en utilisant des nœuds de capteurs intégrés et spécialement conçus pour la construction des maisons intelligentes (voir la figure 2.11). Les nœuds de capteurs peuvent être déployés dans des appareils pour surveiller une tâche : la lumière, température et pression, à titre d'exemple on trouve : les micro-ondes et les machines à laver peuvent être maintenus sans intervention humaine. Les nœuds de capteurs partagent certaines ressources communes comme l'eau, les chauffages et l'électricité dans les projets des maisons intelligentes, par ailleurs, les capteurs contrôlent automatiquement ces appareils et économisent la consommation d'énergie.

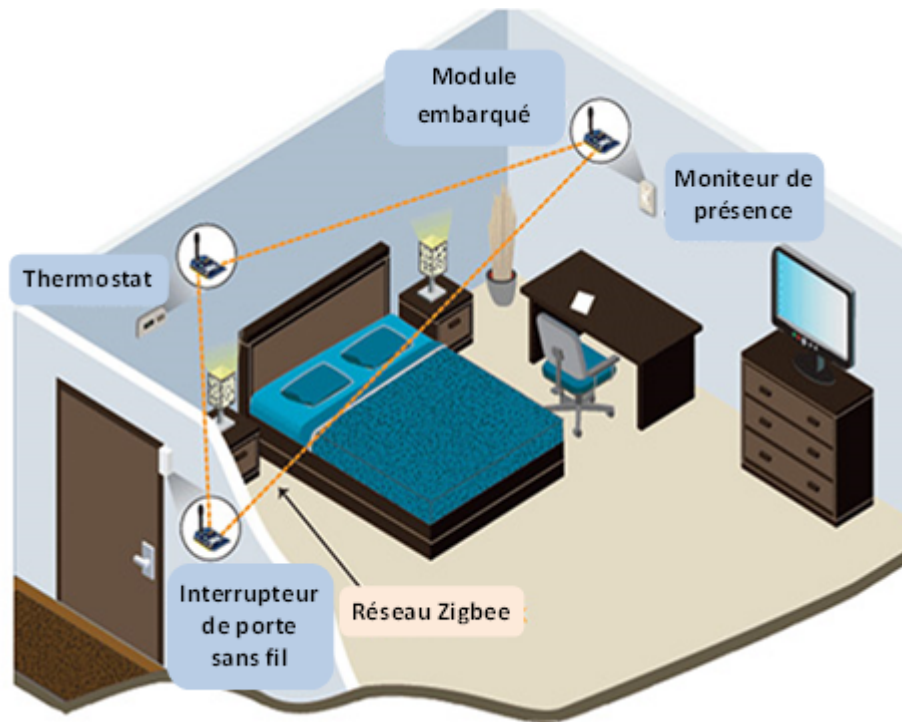


FIGURE 2.11 – Réseau de capteurs sans fil d'une maison intelligente.

## 2.6 LES EXIGENCES DU RCSF

Les exigences d'un bon réseau de capteurs sans fil s'articulent principalement les points fondamentaux suivants :

L'efficacité énergétique, l'évolution, la réactivité, la fiabilité et la mobilité. Un réseau de capteurs sans fil avec ces fonctionnalités peut s'avérer très bénéfique et s'il n'est pas suivi ou assuré, il peut en résulter un réseau qui souffre de surcharge, annulant ainsi son applicabilité.

### 2.6.1 L'efficacité énergétique

L'efficacité énergétique est la capacité des réseaux à gérer les nœuds mobiles et les chemins de données modifiables. La façon dont la conception se déroule est nécessaire pour que le réseau de capteur sans fil soit très réactif pour qu'il puisse gérer la mobilité. [Karimi et Amini, 2019]. Dans la conception, il est nécessaire que le RCSF soit très réactif pour pouvoir gérer la mobilité. En conséquence, il devient plus difficile de concevoir une grande échelle ainsi qu'un RCSF mobile. Avec la mobilité. À la suite, qu'est-ce qu'il devient plus difficile de concevoir une grande échelle, ainsi que des gestes des RCSFs mobiles sont le niveau d'efficacité de l'appareil en question. Ici, c'est la capacité de l'appareil à consommer moins d'énergie pour en faire plus en fonctionnant à des niveaux de puissance extrêmement faibles. Ces appareils sont généralement construits avec la capacité de fonctionner à partir d'une source d'alimentation autre que l'électricité directe. La méthode de conception optimale consisterait à réduire le rapport cyclique de chaque nœud. De plus, les capteurs sans fils sont éga-

lement mises en veille pour économiser de l'énergie et ne répondent donc pas aux communications avec les voisins.

### 2.6.2 L'évolution

La capacité d'un réseau à croître en termes du nombre des nœuds connectés au réseau des capteurs sans fils avec une surcharge excessive peut être qualifiée d'évolution [Alazzawi et al., 2008]. L'implémentation de base d'un tel réseau ne comprend généralement qu'une poignée des nœuds et il est important qu'ils fournissent également un support pour d'autres. D'autant plus les RCSFs sont formés sans aucune topologie prédéterminée et donc les nœuds souhaitant communiquer avec d'autres nœuds doivent générer plus de paquets que ses paquets de données. Ainsi, à mesure que la taille du réseau augmente, plus de paquets seront nécessaires. Les risques de rupture des liaisons de communication sont plus élevés à mesure que la taille du réseau augmente. Ici, il ne restera qu'une petite quantité de bande passante pour la transmission des données de l'application à mesure que le réseau se développe.

### 2.6.3 La réactivité

La capacité du réseau à s'adapter rapidement aux changements de topologie est considérée comme sa réactivité. Il y a cependant des inconvénients à avoir un réseau très réactif [Rajput et Kumaravelu, 2019], des compromis doivent être faits. La latence de la livraison des paquets dans un environnement dynamique ainsi que l'évolution diminueront dans le réseau hautement réactif.

### 2.6.4 La fiabilité

Tout réseau doit être fiable, ce serait l'exigence de base. Vous avez besoin d'une transmission de données fiable dans un état de changement continu de la structure du réseau. Il existe généralement une relation inverse entre l'évolution et la fiabilité dans les réseaux sans fils ad hoc [Damaso et al., 2014]. En effet, à mesure que le nombre de nœuds du réseau augmente, il devient plus difficile de maintenir la fiabilité. Si le réseau est hautement évolutif et mis à l'échelle dans un réseau plus grand que ce qu'il était initialement conçu pour être alors cela mettra une pression sur la fiabilité de la transmission des données et le point de rupture apparaîtra plus tôt.

### 2.6.5 La mobilité

La mobilité est la capacité des réseaux à gérer les nœuds mobiles et les chemins des données modifiables [Maheswar et al., 2019]. Dans la conception, il est nécessaire qu'un RCSF soit très réactif pour pouvoir gérer la mobilité. À la suite de quoi il devient plus difficile de concevoir une grande échelle ainsi qu'un réseau de capteurs sans fil mobile.

## 2.7 LES CARACTÉRISTIQUES DU RCSF

Une caractéristique importante des applications du réseau de capteurs sans fil est la capacité à installer facilement un grand nombre de nœuds de capteurs sans fil. Cette fonctionnalité déclenche tous les défis de conception et d'implémentation qui entourent normalement la communication sans fil, ainsi que d'autres défis propres aux applications particulières. Les principales caractéristiques sont l'efficacité énergétique, les interférences, la sécurité, la gestion des données et le déploiement à grande échelle. La conception et la mise en œuvre des réseaux de capteurs sans fils doivent traiter tous ces problèmes [Yong-Min et al., 2009] [Paik et al., 2009] [Ayaz et al., 2018].

### 2.7.1 Coût de production

Cette contrainte générale dans la conception du réseau de capteurs est de fournir le coût du nœud de capteur le plus bas possible. Les RCSFs sont constitués d'un grand nombre de capteurs pour calculer ou estimer le coût du réseau, tandis que les dépenses du nœud de capteur particulier sont plus importantes. Ainsi, le coût du nœud de capteur devient aussi difficile avec toutes les exigences de l'application.

### 2.7.2 Absence d'infrastructure

les réseaux de capteurs en particulier se distinguent des autres réseaux par la propriété d'absence d'infrastructure préexistante et de tous genres d'administration centralisée.

### 2.7.3 Evolution

Le nombre des nœuds dans un réseau de capteurs sans fil est généralement élevé. Cependant, l'échelle dépend en partie de la zone couverte, de la réplique des nœuds, de la couverture de détection limitée et des exigences de l'application. Par conséquent, les protocoles du réseau de capteurs sans fil devraient être capables de gérer le nombre des nœuds et également des densités élevées.

### 2.7.4 Interférences

Les liens radios ne sont pas isolées, deux transmissions simultanées sur une même fréquence, ou utilisant des fréquences proches, peuvent interférer. Les interférences causées par d'autres systèmes sans fils fonctionnant sur une bande de fréquences similaires et coexistants dans le même voisinage peuvent réduire considérablement les performances des RCSFs. Le mécanisme d'évitement des interférences ordinaires peut ne pas être efficace pour un RCSF à grande échelle en raison des contraintes des RCSFs telles que sa faible capacité de calcul.

### 2.7.5 Topologie du réseau

La conception géographique d'un réseau de capteurs est appelée topologie du réseau, il doit contenir une valeur de haute densité des nœuds de capteurs, ces capteurs peuvent être attachés à des objets mobiles, ces capteurs déplacent d'une façon libre ainsi la topologie du réseau demandé d'être adapté au changement et aux dynamicités.

### 2.7.6 La sécurité physique limitée

les réseaux de capteurs sans fils sont plus touchés par le paramètre de sécurité que les réseaux filaires classiques. Cela se justifie par les contraintes et les limitations physiques qui font que le contrôle des données transférées doivent être minimisées.

### 2.7.7 la bande passante limitée

une des caractéristiques primordiales des réseaux basés sur la communication sans fil est l'utilisation d'un médium de communication partagé. Ce partage fait que la bande passante d'un nœud est limitée.

### 2.7.8 Le stockage et la récupération

Les réseaux de capteurs étudient les phénomènes environnementaux et fabriquent une grande quantité de données brutes dans une série chronologique continue. Bien que les données soient continuées, la gestion de base de données n'est pas adaptée aux réseaux de capteurs sans fils.

### 2.7.9 La durée de vie

La durée de vie du réseau est un problème sérieux dans les réseaux de capteurs sans fils. Elle est considérée comme la caractéristique la plus critique dans les réseaux de capteurs sans fils. Les nœuds sont alimentés par batterie ou l'énergie est récupérée de l'environnement et leur maintenance est également difficile. Ils devraient être parfaitement fonctionnels pendant de longues périodes. Ainsi, l'équilibrage de la charge et les économies d'énergie devraient être pris en compte dans la mise en plateformes, applications et les protocoles des réseaux de capteurs sans fils [Ayadi et al., 2018].

## 2.8 LES PROBLÈMES D'OPTIMISATION DANS LES RCSFs

En science et technologie, l'optimisation concerne l'utilisation efficace des ressources disponibles avec les facteurs souhaités et indésirables, pris en compte de manière à ce que les meilleures performances puissent être obtenues par le système.

Dans les RCSF, les ressources telles que l'énergie du nœud, la fréquence, la mémoire de stockage et l'unité de calcul du nœud de capteur sont des facteurs critiques. Les problèmes d'optimisations sont basés sur une fonction objective et des contraintes qui sont associées. Le but de la fonction objective est de minimiser ou de maximiser les objectifs, qui peuvent être

classés comme des fonctions à objectif unique et des fonctions à objectifs multiples. Les contraintes liées aux formes des variables sont utilisées pour définir la frontière de la solution à une plage de solution. La définition formelle du problème d'optimisation (minimisation) ayant  $n$  variables et  $m \geq 1$  objectif peut-être défini comme :

$$\text{Min } f_i(x) (i = 1, 2, \dots, l)$$

La fonction  $f_i$  est appelée fonction objective (fonction du coût), en outre, lorsque  $i$  est égale à 1, alors on parle de fonctions à objectif unique. Mais lorsque :

$$\text{Min } g_i(x) (i = 1, 2, \dots, m)$$

$$\text{Min } g_i(x) (i = 1, 2, \dots, n)$$

L'optimisation est multiobjectif [Sawaragi et al., 1985], [Taha, 2020]. Dans le contexte des problèmes d'optimisations, les propriétés et les dérivées de la variable et des équations sont utilisées pour effectuer une catégorisation supplémentaire des problèmes d'optimisations, tels que linéaire, non linéaire, quadratique, etc. pour des problèmes simples et multiobjectifs. Par ailleurs, lors de l'optimisation des problèmes des RCSFs, il est nécessaire de considérer à la fois le problème et la solution. Le type de problème et de la solution dépendra des facteurs tels que l'objectif d'optimisation ou le modèle du réseau.

### 2.8.1 Conception et déploiement

La nature omniprésente des réseaux de capteurs sans fils est utile pour effectuer des mesures dans des environnements difficiles et inaccessibles de manière efficace. Plusieurs techniques peuvent être très efficaces pour concevoir et planifier le déploiement des nœuds dans tels environnements. Le problème de conception et du déploiement du RCSF fait référence au placement optimal des nœuds et des stations de bases afin de garantir la couverture et la connectivité avec une efficacité énergétique adéquate sont atteints [Bojkovic et Bakmaz, 2008]. Dans certains cas, les nœuds de capteurs qui doivent être déployés sont déterminés à l'avance, comme dans les applications de surveillance de la santé (suivie des patients), par contre, dans la surveillance des catastrophes, un tel positionnement n'est pas pratique et ils sont déployés de manière ad hoc (sans infrastructure). Alors, le déploiement optimal des capteurs peuvent garantir une qualité de service adéquat, une durée de vie prolongée et une communication sécurisée [Kulkarni et Venayagamoorthy, 2010]. Le déploiement des nœuds dans le RCSF est basé sur des nœuds stationnaires ou mobiles [Aziz et al., 2007], des travaux de recherche ont travaillé sur la minimisation de la surface des trous de couverture via un algorithme d'optimisation PSO. PSO et diagramme de Voronoi ont été proposés pour résoudre le problème de couverture causée par le nombre des capteurs. L'idée est basée sur le principe que si un capteur couvre tous les points de la région d'intérêt, alors toute la région d'intérêt est couverte. Le diagramme de Voronoi est utilisé pour évaluer l'adéquation de la couverture du RCSF. Basé sur ceci fitness, l'algorithme recherche la position la plus optimale des capteurs. Ce schéma trouve une couverture proche d'optimale mais ignore la complexité de la détermination des polygones de Voronoi [Tillett et al., 2002].

### 2.8.2 Agrégation de données

L'agrégation de données est un processus de collection et de présentation des données dans un format résumé. Les données peuvent être collectées à partir de plusieurs sources de données dans le but de combiner ces sources de données dans un résumé pour l'analyse des données. Il s'agit d'une étape cruciale car la précision des informations issues de l'analyse des données dépend fortement des quantités et de la qualité des données utilisées. Il est important de recueillir des données précises de haute qualité et une quantité suffisante pour créer des résultats pertinents. Lorsque le RCSF a pour but de surveiller une région définie, chaque nœud de capteur du réseau collecte des observations locales et envoie ces données compressées ou partiellement traitées (un résumé) au centre de fusion. Le centre de fusion (centre d'agrégation de données) utilise le résumé et par la suite applique une règle de fusion de décision spécifique pour prendre la décision finale. Le principal objectif de l'agrégation de données est de rassembler et d'agréger les données de manière économe en énergie afin que le réseau aura une durée de vie prolongée. La fusion de données est un processus distribué et répétitif qui convient parfaitement aux approches métaheuristiques. L'agrégation efficace des données influence la performance du réseau par conséquent, il est raisonnable de choisir une meilleure technique pour contrôler les paramètres de fusion. Plusieurs approches ont fourni l'optimisation dans plusieurs aspects de l'agrégation de données. Un schéma descriptif d'agrégation des données est présenté dans la figure 2.12.

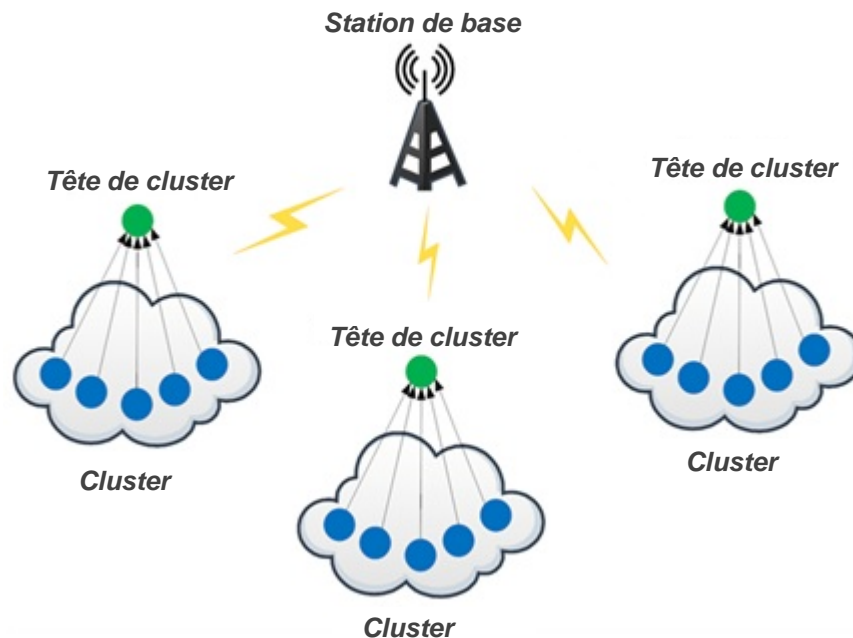


FIGURE 2.12 – Agrégation de données dans des RCSFs.

### 2.8.3 Protocole de routage

Le protocole de routage est un processus permettant de sélectionner un chemin approprié pour que les données déplacent de la source jusqu'à la destination. Le processus rencontre plusieurs difficultés lors de la sélection de l'itinéraire, qui dépend du type de réseau, des caractéristiques du canal et des mesures de performance. Les données détectées par les nœuds de capteurs dans le RCSF sont généralement transmises à la station de base qui connecte le réseau de capteurs aux autres réseaux (via Internet) où les données sont collectées, analysées et des mesures sont prises en conséquence. Dans les réseaux de capteurs de petite taille où la station de base communique directement avec les nœuds de capteurs via un saut unique (direct), donc on parle du paradigme centralisé, donc toutes les données sont contrôlées directement par la station de base, mais dans la plupart des applications du RCSF, dans le cas d'un RCSF à grande échelle où des milliers de nœuds doivent être placés, ce scénario nécessite une communication multi-sauts, car la plupart des nœuds de capteurs sont si loin du nœud récepteur (passerelle) qu'ils ne peuvent pas communiquer directement avec la station de base ou la communication consomme plus d'énergie, cette communication à plusieurs sauts s'appelle communication multi-sauts (indirecte) [CS et al., 2011]. Dans une communication indirecte les nœuds capteurs non seulement produisent et livrent leur matériel, mais détermine également la trajectoire du chemin pour d'autres nœuds de capteurs vers la station de base. Le processus de recherche d'un chemin approprié du nœud source au nœud de destination s'appelle routage et c'est la responsabilité principale de la couche réseau. La tâche de conception des protocoles de routages est assez difficile en raison des multiples caractéristiques qui les différencient des réseaux sans infrastructure. Plusieurs types de challenge de routage impliqués dans les réseaux de capteurs sans fils. Les défis essentiels sont résumés comme suit :

- 1 - Il est presque difficile d'allouer un schéma d'identificateurs universels pour une grande quantité de nœuds de capteurs. Les capteurs sans fils ne maîtrisent donc pas les protocoles IP classiques.
- 2 - Le flux des données détectées est obligatoire depuis plusieurs sources vers une station de base spécifique. Mais cela ne se produit pas dans les réseaux de communications typiques.
- 3 - Le trafic de données créées présente une redondance importante dans la plupart des cas, étant donné de nombreux nœuds de détection peuvent générer les mêmes données lors de la détection. Il est donc essentiel d'exploiter cette redondance et d'utiliser la bande passante et l'énergie disponibles aussi efficacement que possible.
- 4 - De plus, les moteurs sans fils sont fermement limités dans les relations entre l'énergie de transmission, la bande passante, la capacité et le stockage et l'énergie embarquée. En raison de ces différences, un certain nombre de nouveaux protocoles de routage ont été projetés afin de faire face à ces défis de routage dans les réseaux de capteurs sans fils.
- 5 - Quelques algorithmes métaheuristiques ont été proposées pour choisir d'itinéraire optimale des circulations des données dans les réseaux de capteurs sans fils, recherche taboue [Semchedine et al., 2011], une approche hybride : basée sur l'algorithme de recherche coucou et l'algorithme de

recherche d'harmonie [Gupta et Jha, 2018], l'algorithme d'essaim de poissons artificiels et l'algorithme des colonies de fourmis [X et al., 2018].

6 - Un schéma descriptif du protocole de routage est présenté dans la figure 2.13.

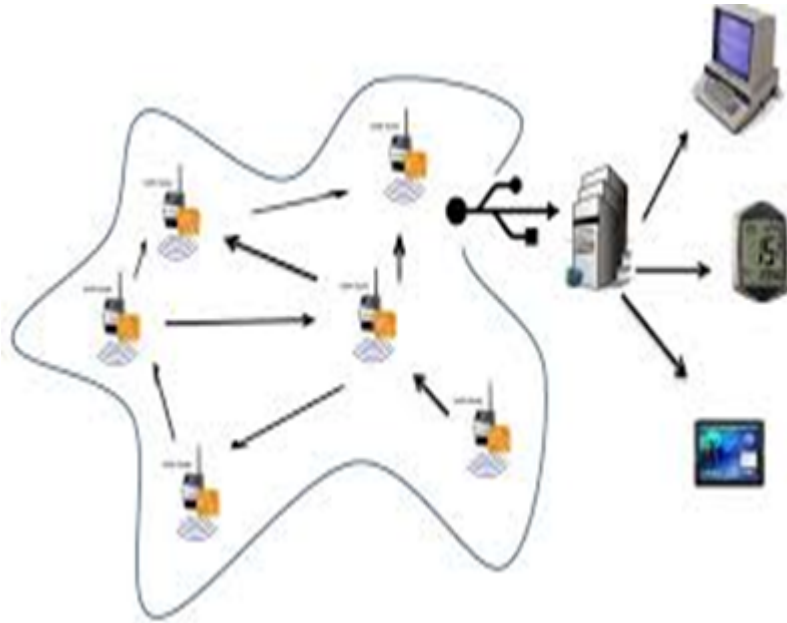


FIGURE 2.13 – Protocole de routage dans le RCSF.

#### 2.8.4 Clustering

Le clustering (regroupement) est une tâche importante dans les réseaux de capteurs sans fils pour l'efficacité énergétique et la stabilité du réseau (durabilité). Le regroupement dans les RCSFs est bien étudié et plusieurs approches ont été proposées comme solution. Actuellement, le clustering a connu un progrès très rapide pour faire face aux problèmes réels tels que la durée de vie du réseau, la consommation d'énergie et le problème d'évolution. Les algorithmes de clustering limitent la communication dans un domaine local et n'envoient que les informations nécessaires au reste du réseau via les nœuds de transfert (nœuds de passerelle). Un groupe de nœuds forme un cluster et les interactions locales entre les membres du cluster sont contrôlées via une tête de cluster (Cluster Head), voir la figure 2.14. Les membres du cluster communiquent généralement avec la tête de cluster et les données collectées sont agrégées et fusionnées par la tête de cluster pour conserver l'énergie. Les têtes de cluster peuvent également former un autre niveau de cluster avant d'atteindre la station de base.

Une hiérarchie du clustering de basse consommation d'énergie (LEACH : low energy-aware clustering hierarchy) est proposée dans [Heinzelman et al., 2000], c'est un algorithme simple et efficace, le clustering est défini comme un problème NP difficile d'où plusieurs algorithmes méta-heuristiques ont été proposés, l'algorithme PSO est appliqué dans le but de créer un équilibre entre le nombre de nœuds et des candidats

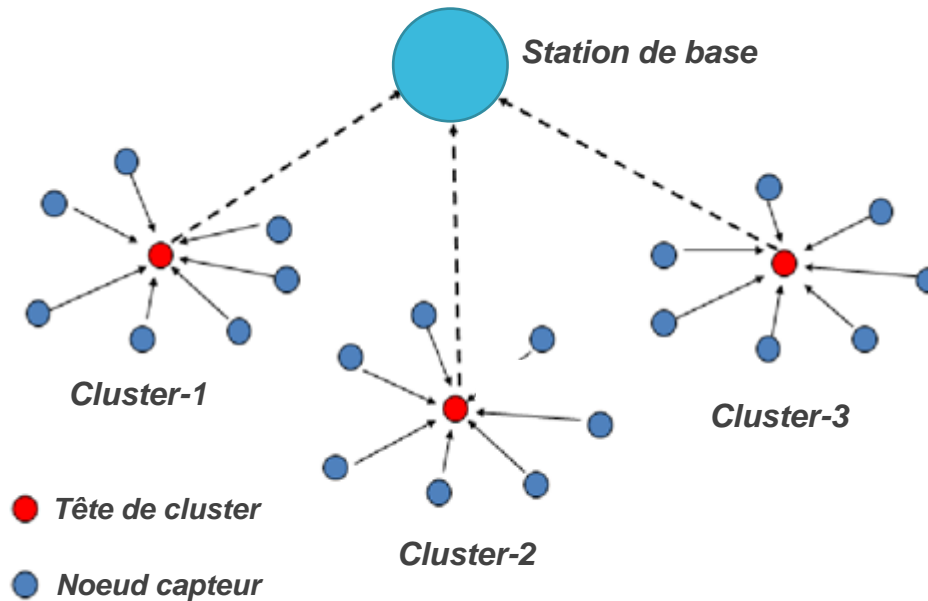


FIGURE 2.14 – Clustering dans le RCSF.

CH dans chaque cluster d'un réseau [Tillett et al., 2002], il a pour objectif de minimiser l'énergie dépensée par les nœuds tout en maximisant la transmission de données. Cependant, aucune comparaison avec d'autres stratégies de regroupement de référence n'a été abordée. Une approche hybride basée sur l'algorithme des colonies de fourmis (ACO) et l'algorithme des essaims de particules (PSO), elle est proposée comme protocole de regroupement éco-énergétique et de routage [Kaur et Mahajan, 2018]. Initialement, les clusters sont formés sur la base de l'énergie restante, puis l'agrégation de données est basée sur l'approche hybride entrera en exécution pour améliorer encore l'agrégation des données inter-clusters.

### Objectifs du Clustering

le clustering cherche à optimiser une tâche multi-objective, les principaux objectifs sont comme suit :

- 1 - Permet l'agrégation.
- 2 - Minimiser la transmission des données.
- 3 - Facilité de réutilisation des ressources.
- 4 - Les têtes du cluster (cluster-head) et les nœuds de passerelle peuvent former un axe principal pour le routage des inters-clusters.
- 5 - La structure du cluster donne l'effet d'un réseau plus petit et plus stable.
- 6 - Prolonger la durée de vie du réseau.
- 7 - Minimiser le trafic du réseau et le litige pour le canal.
- 8 - L'agrégation et les mises à jour des données ont lieu dans les têtes de cluster.

### 2.8.5 Qualité de service

La prise en charge de la qualité de service dans les réseaux de capteurs sans fils est difficile en raison des contraintes énergétiques et de ressources de calcul sévères des capteurs sans fils. De plus, certaines propriétés de service telles que : Le taux de paquets perdu, le débit, bande passante, la gigue, le délai, la fiabilité, la durée de vie du réseau et la qualité des données peuvent entrer en conflit par nature.

Le routage à chemins multiples, par exemple, peut améliorer la fiabilité, cependant, il peut augmenter la consommation d'énergie et les retards dus aux transmissions en double. De plus, les lectures du capteur haute résolution entraînent plus de consommations d'énergie et de retards. Modéliser ces relations, mesurer la qualité fournie et fournir des moyens de contrôler l'équilibre est essentiel pour la prise en charge de la qualité de service [Singh et Kumar, 2018].

En outre, la qualité de Service (QoS) est définie comme des mesures de la fiabilité des applications dans un objectif d'efficacité énergétique. Dans les applications en temps réel, la qualité de service permet aux données détectées d'être livrées dans un délai limité. La recherche de QoS se concentre depuis plusieurs années sur ces paramètres d'évaluations La qualité de service (QoS) est très demandée pour un nombre d'applications du RCSF tels que les réseaux de zone corporelle, les réseaux ad hoc de véhicules (en anglais VANET), le suivi et la surveillance des cibles militaires [Takale et Lokhande, 2018], etc. L'obtention d'une meilleure (satisfaite) qualité de service dans ces réseaux fortement limités en ressources n'est pas une tâche facile. Dans un certain nombre de cas, les métriques ou paramètres de QoS peuvent même entrer en conflit avec eux-mêmes. À titre d'exemple, presque toutes les applications médicales, la rapidité ou l'information temps réel (livraison de données) est obligatoire, mais cela peut entrer en conflit avec l'efficacité énergétique (se considère comme un paramètre de QoS), l'utilisation de l'optimisation est donc nécessaire dans tous ces scénarios de QoS conflictuels. Comme le QoS et l'efficacité énergétique.

### 2.8.6 Localisation

Les RCSFs sont étroitement associés aux phénomènes physiques dans leur environnement. Les informations recueillies doivent être associées à l'emplacement des nœuds de capteurs pour fournir une vue précise du champ de capteur observé. De plus, les RCSFs peuvent être utilisés pour suivre certains objets pour des applications de surveillance, ce qui nécessite également que les informations de localisation des nœuds de capteurs soient incorporées dans les algorithmes de suivi durant la mobilité, les protocoles de routage géographique, qui fournissent une évolution et des décisions localisées pour la communication à sauts multiples [Boukerche et al., 2007], cela nécessite de connaître l'emplacement des nœuds. Ces exigences motivent le développement des protocoles de localisation efficaces pour les RCSFs. Les protocoles de localisation peuvent être classés en deux grandes catégories :

1. Protocoles de localisation basées sur la plage.
2. Protocoles de localisation sans plage.

La localisation des nœuds est très demandé pour un certain nombre d'ap-

applications de RCSF telles que les réseaux de zone corporelle, les réseaux ad hoc de véhicules (VANET), le domaine de la santé (suivi des patients) et la surveillance des cibles militaires (détection des ennemis), domaine environnemental (détection des feux, détection des inondations) etc. alors quelques domaines d'applications sont présentées dans la figure 2.15.

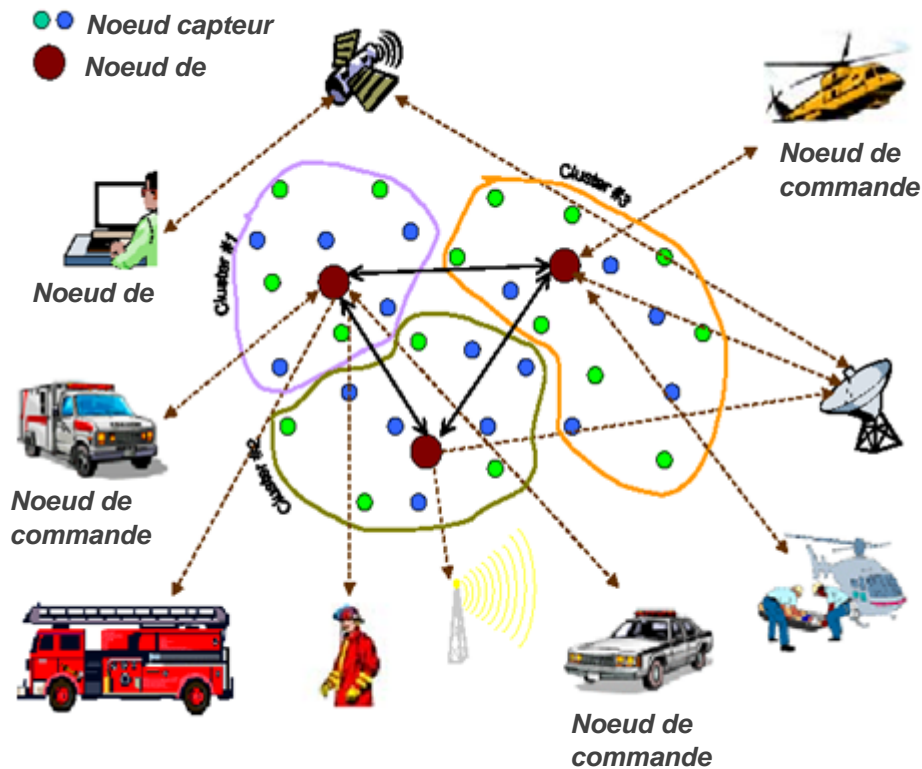


FIGURE 2.15 – La localisation dans les RCSFs.

La localisation basée sur la plage nécessite l'existence des nœuds de balises disposant des informations de localisation précise. En utilisant plusieurs techniques de télémétrie, les nœuds restants du réseau estiment leur distance à trois nœuds de balise ou plus. Sur la base de ces informations, l'emplacement d'un nœud est estimé. Par contre, les protocoles de localisation sans portée, ne nécessitent pas d'estimation de distance. Bien que l'existence d'un nœud de balise puisse encore être requise, l'emplacement des autres nœuds est estimé par une technique sans plage...

Plusieurs algorithmes ont été proposés pour résoudre le problème de la localisation [Paul et Sato, 2017] et plusieurs techniques ont été abordées [Mohar et al., 2018].

## 2.9 CONCLUSION

Les réseaux de capteurs sans fils (RCSFS) représente une technologie d'actualité couvrant le domaine d'application générale de la surveillance et également représentent un domaine de recherche intéressant en raison de leurs nombreuses applications et de leur intégration vers des systèmes de réseaux plus complexes.

Les RCSFs se composent généralement de nombreux appareils (nœuds de capteurs) pour permettre une surveillance à grande échelle. Les principales exigences des réseaux de capteurs sans fils doivent être de petite taille et à faible coût. Ainsi, au lieu d'implémenter des applications complexes dans le réseau lui-même, le réseau de capteurs effectue deux opérations principales : détecter et communiquer les données détectées entre elles et avec un serveur central ou station de base directement. Les applications des réseaux de capteurs sans fils couvrent la surveillance environnementale et animale, la surveillance industrielle, la surveillance militaire, la surveillance de la santé des patients et de nombreux autres domaines. L'une des principales caractéristiques des réseaux de capteurs sans fils est qu'ils sont étroitement liés à leur application. Un système développé pour la surveillance industrielle dans un endroit peut être difficile de l'utiliser dans un autre endroit et est presque impossible à utiliser dans la surveillance des patients. Par ailleurs, les difficultés des RCSFs sont généralement liées à leurs contraintes strictes telles que l'énergie, la bande passante et la mémoire. Dans ce chapitre, nous avons défini et décrit les notions fondamentales des réseaux de capteurs sans fils, l'architecture physique d'un capteur sans fil et ses fonctionnalités. Nous avons mis l'accent sur les exigences et les caractéristiques des réseaux de capteurs, on a cité brièvement le problème de la localisation et les contraintes liées aux données qui y transitent. Ainsi, nous avons décrit certains problèmes fondamentaux liés au déploiement, au protocole de routage, la localisation, l'agrégation des données et à la qualité de service, ces problèmes la sont généralement modélisés comme des problèmes d'optimisations autonomes ou intégrées dans la conception des protocoles. Dans le chapitre suivant, nous nous sommes ensuite intéressés et concentrera au contexte applicatif qui n'est autre que la localisation des nœuds.

Un état de l'art détaillé de l'évolution du problème de localisation des nœuds est élaboré avec une étude comparative introduite, ainsi, un schéma très performant proposé avec évaluation de performances.

# LA LOCALISATION DES NŒUDS DANS LES RCSFs

# 3

## 3.1 INTRODUCTION

L'une des applications cruciales des réseaux sans fils consiste à déterminer l'emplacement physique des objets à tout moment donné. La connaissance en temps réel de l'emplacement du personnel, des actifs et des instruments portables peut augmenter l'efficacité de la gestion. L'estimation d'emplacement fait référence au processus d'obtention des informations d'emplacement sur un nœud par rapport à un ensemble de positions de référence connues. L'estimation d'emplacement est également appelée positionnement, localisation et géolocalisation. La connaissance de l'emplacement des nœuds offre la possibilité de fournir des services dépendant de l'emplacement. Par exemple, un patient dans une zone couverte peut transporter via un appareil audio/vidéo qui fournit des informations pertinentes au médecin traitant, en fonction de son emplacement dans le service de l'hôpital ou un endroit spécifique. L'emplacement d'un nœud peut également être utilisé dans le cadre du processus d'authentification. De cette façon, l'authenticité d'un paquet est déterminée non seulement par les informations incorporées dans le paquet, mais également par l'emplacement du nœud qui a transmis le paquet.

Ici, nous nous concentrons sur les méthodes d'estimation d'emplacement qui utilisent des signaux de radiofréquence (RF : radiofrequency). Cependant, il est possible d'utiliser d'autres types de signaux tels que des ultrasons ou des infrarouges au lieu d'un signal radiofréquence dans un algorithme d'estimation de localisation, mais les systèmes de positionnement basés sur radiofréquence s'avèrent également plus adaptés à une grande échelle déploiements. À titre d'exemple, L'indicateur de force du signal reçu (RSSI) est une mesure de la qualité du signal radiofréquence qui est souvent utilisé dans le domaine de la localisation spatiale. Les systèmes d'estimations de localisation développées à l'aide du réseau sans fil sont parfois appelés systèmes de positionnement local (LPS) pour les différencier des systèmes de positionnement globaux (GPS). Un appareil compatible GPS détermine sa position en calculant sa distance par rapport à trois nœuds GPS. Chaque nœud transmet de manière continue un message contenant l'emplacement du satellite et l'heure exacte. Le récepteur GPS compare l'heure exacte à laquelle le message a été reçu et l'heure à laquelle le message a été transmis par le satellite pour calculer la distance parcourue. Connaissant la distance d'au moins trois satellites et la position

des satellites, le récepteur calcule sa propre position. Le LPS, en revanche, n'utilise pas les informations fournies par les satellites GPS ou tout autre émetteur long porté. Un LPS utilise les signaux RF transmis par des nœuds locaux dont les positions sont connues ou le nœud mobile lui-même pour calculer l'emplacement du nœud mobile par rapport aux emplacements connus des autres nœuds locaux. Le choix de l'algorithme d'estimation de localisation dépend du scénario d'application. Les méthodes d'estimation de localisation sont comparées en fonction de leurs performances et de leur complexité. La précision d'emplacement, qui est la distance entre l'emplacement réel et l'emplacement estimé, est la mesure de performance la plus intuitive.

L'estimation d'emplacement implique généralement deux groupes de nœuds. Le premier groupe se compose de nœuds avec des emplacements connus. Ces nœuds, parfois appelés nœuds d'ancrage ou référence, sont utilisés comme références pour l'estimation de l'emplacement. L'emplacement des nœuds d'ancrage peut être déterminé par l'installateur, ou les nœuds d'ancrage peuvent être équipés de GPS pour déterminer leur propre emplacement. Le deuxième groupe est constitué des nœuds dont les emplacements sont inconnus, appelés nœuds à localiser ou suivis ou cibles. Le but de l'estimation d'emplacement est de déterminer l'emplacement des nœuds suivis à l'aide des nœuds d'ancrage. L'idée de base du positionnement local peut être résumée comme suit :

Un nœud à localiser avec un emplacement inconnu émet un signal, qui est reçu par les nœuds d'ancrage voisins. Les nœuds d'ancrage mesurent l'indicateur de la force du signal reçu (RSSI), l'heure d'arrivée (ToA), ou l'angle d'arrivée (AoA) du signal reçu. Ces valeurs mesurées sont utilisées comme entrées pour un algorithme qui détermine l'emplacement approximatif du nœud suivi en se basant sur des méthodes de calcul de position comme trilatération ou angulation. Ces algorithmes utilisent normalement une seule de ces trois entrées. La dernière étape est de fait appel aux algorithmes de localisation pour déterminer les emplacements exacts des nœuds inconnus détectés. Il existe deux types d'approches de traitement pour l'estimation de la position des nœuds. Ces approches sont des approches de traitement centralisées et distribuées. Dans l'approche de traitement central, un nœud unique appelé nœud de traitement d'emplacement central ou station de base ou station centrale est dédié à l'exécution de l'algorithme d'estimation d'emplacement. Tous les autres nœuds du réseau uniquement recueillir les informations liées à l'emplacement tel que RSSI et les envoyer au nœud de traitement d'emplacement central. Le nœud de traitement d'emplacement central calcule l'emplacement estimé de tous les nœuds suivis et communique l'emplacement calculé à chaque nœud suivi s'il sera demandé, par contre, dans une approche de traitement distribué, la tâche de la localisation est répartie entrées presque tous les nœuds du réseau. De cette façon, il n'y a pas de nœud de traitement d'emplacement centralisé et chaque nœud détermine son propre emplacement en communiquant uniquement avec les nœuds d'ancrage voisins et potentiellement d'autres nœuds suivis.

Par ailleurs, des travaux de recherche ont mis aux points des diverses approches, selon la nature du problème donné.

Ces approches différentes dont les hypothèses qu'elles font sur leurs

capacités respectives du réseau et du périphérique. Ceux-ci sont liés au matériel de l'appareil, aux modèles de propagation du signal, aux exigences de synchronisation et d'énergie, à la composition du réseau (homogène vs hétérogène), à la nature de l'environnement (intérieur vs extérieur), à la densité des nœuds ou nœuds ancres, d'autre aux exigences de synchronisation, aux coûts de communication, les exigences d'erreur et la mobilité de l'appareil. Tous ces challenges-là et malgré les efforts de recherche, de nombreux problèmes restent ouverts dans le domaine de la localisation.

## 3.2 PROBLÈME DE LA LOCALISATION :

### 3.2.1 Contexte général

Les informations de localisation sont utilisées pour détecter et enregistrer des événements, ou pour acheminer des paquets à l'aide d'un routage géométrique [Anani et al., 2019] [Guan et al., 2019]. Un réseau de capteurs sans fil est un ensemble de  $N$  nœuds de capteurs, déployé dans une zone de détection  $Z$  de  $K$ -dimension, où  $K = [0, s] \times [0, s]$ , chaque nœud peut détecter l'évènement via un rayon de communication  $R$ .

Nous considérons la liaison de communication entre les nœuds n'est pas symétrique, c'est-à-dire que pour deux nœuds  $U$  et  $V$  quelconques, le nœud  $U$  communique avec le nœud  $V$  si et seulement si le signal de  $V$  arrive au nœud  $U$  avec la même puissance de signal  $W$ . Ainsi, nous représentons le réseau par le graphe euclidien  $G = (V, E)$  avec les propriétés suivantes :

1.  $V = \{V_1, V_2, \dots, V_n\}$  est l'ensemble des nœuds de capteurs.

2.  $(i, j) \in E$ , si  $V_i$  communique avec  $V_j$ , i.e :

$$\begin{cases} (i, j) \in E & V_{ij} < R \\ (i, j) \notin E & \text{Sinon} \end{cases}$$

3.  $W(e) \leq R$  est le poids de l'arête,  $e = (i, j)$  est la distance entre  $V_i$  et  $V_j$ . Dans un réseau de capteurs, les nœuds peuvent être classés comme :

**Nœuds inconnus (U)** : également appelé nœuds libres ou muets, ce terme fait référence aux nœuds du réseau qui ne connaissent pas leurs informations de localisation. Permettre à ces nœuds d'estimer leur position qui est l'objectif principal d'un système de localisation.

**Nœuds installés (établi) (S)** : ces nœuds étaient initialement des nœuds inconnus qui ont réussi à estimer leurs positions en utilisant le système de localisation. Le nombre de nœuds installés et l'erreur de position estimée de ces nœuds sont les principaux paramètres pour déterminer la qualité d'un système de localisation.

**Nœuds d'ancre (Balise ou référence) (B)** : aussi appelés points de repère ou ancres, ce sont les nœuds qui n'ont pas besoin d'un système de localisation pour estimer leurs positions physiques. Leur localisation est obtenue par placement manuel ou par des moyennes externes telles que le GPS. Ces nœuds forment la base de la plupart des systèmes de localisation pour les RCSFs.

### 3.2.2 Formulation du problème

Il convient de noter que le problème de la localisation n'est généralement résolu que pour deux dimensions, en supposant qu'en cas de besoin ou de déploiement, il pourrait être étendu à trois dimensions. C'est pour cette raison, nous avons déclaré que le graphe  $G$  est bidimensionnel. Par conséquent, on peut dire que  $G$  est un graphe euclidien dans lequel chaque nœud de capteur a une coordonnée dans un espace à deux dimensions. La coordonnée  $(x_i, y_i)$  représente l'emplacement d'un nœud  $i$  dans le champ de capteur donné.

soit un réseau de capteurs à sauts multiples représenté par un graphe  $G = (V, E)$

Le graphe a un ensemble de nœuds ( $B$ ) représente l'ensemble de nœuds ancre avec des positions connues données par  $(x_b, y_b)$ ,  $\forall b \in B$ , trouver la position  $(x_u, y_u)$  d'autant  $u \in U$  que possible, Si  $u$  est trouvé alors  $S(u)=U(u)$ .

Le problème du positionnement ou géolocalisation ou de la localisation des nœuds dans un réseau de capteurs sans fil peut être résolu si chaque nœud est équipé par un système de positionnement global GPS. Cependant, dans le cas des réseaux de capteurs, cette option n'est pas envisageable pour les raisons suivantes :

1 - Le récepteur GPS et les protocoles utilisés ne sont pas conçus pour être énergétiques. Dans le cas des réseaux de capteurs sans fils, l'énergie est une ressource rare et les nœuds de capteurs peuvent être déployés sans qu'aucune sorte de batterie ne soit pas remplacée pendant de nombreuses années. Par conséquent, les GPSs ne conviennent pas à la solution du problème de localisation dans les réseaux de capteurs sans fils. Il est néanmoins possible que les nœuds ancres qui ne constituent qu'une minorité du nombre total des nœuds soient équipés d'un GPS afin que ceux-ci puissent servir des nœuds de référence à d'autres nœuds pour résoudre le problème de leurs connaissances de position à l'aide de l'algorithme de localisation [Guan et al., 2019].

2 - Les appareils GPSs sont beaucoup plus chers. Si chaque nœud de capteur sera équipé par un GPS alors le coût de déploiement peut augmenter dans une certaine mesure afin de rendre la solution de réseau de capteurs impossible pour un problème particulier [jie et al., 2008]. [Khoufi et al., 2017]

3 - L'une des propriétés requises des nœuds de capteurs est qu'ils doivent être de petite taille. Avec l'ajout d'un appareil GPS, la taille des nœuds de capteurs deviendrait assez grande, ce qui violerait à nouveau l'une des principales exigences d'un nœud de capteur [Dinh et Kim, 2019].

4 - Les appareils GPSs dépendent des satellites pour leur fonctionnement. Dans les cas ou dans des circonstances où aucune liaison satellite n'est pas disponible, le GPS cesse de fonctionner. Dans certaines applications, cela peut vraiment être le cas par exemple. Dans un terrain montagneux, souterrains ou autres endroits avec une ligne de vue bloquée communiquant avec les GP Ss.

Pour les raisons décrites ci-dessus, les appareils GPSs ne sont utilisés que dans une minorité des nœuds qui servent de nœuds de référence pour résoudre le problème de localisation des autres nœuds. Ces nœuds sont éga-

lement appelés nœuds de balise. Alternativement, il est possible d'éviter complètement l'utilisation du GPS en positionnant quelques nœuds à des points fixes afin que leur position soit connue a priori afin que ces nœuds puissent servir de nœuds de référence. Le champ du capteur peut ensuite exploiter la capacité inhérente aux radiofréquences des nœuds de capteurs ou autres techniques pour déterminer leurs positions à l'aide d'un algorithme de localisation. La précision avec laquelle les nœuds inconnus (à localiser) peuvent déterminer leur emplacement dépend de la portée de transmission des nœuds de référence et de la distance entre deux nœuds de référence adjacents [Mass-Sanchez et al., 2017] [Hongyang Chen et al., 2008], c'est-à-dire la densité des nœuds de référence influence le taux de localisation.

### 3.3 OBJECTIF DE LOCALISATION

La localisation dans les RCSFs où les nœuds sont aléatoirement déployés consiste à déterminer les coordonnées géographiques des différents capteurs dans un temps optimal et avec minimal d'énergie consommée. La localisation est nécessaire, non seulement pour localiser les différents événements survenus dans la zone surveillée, mais aussi pour le développement des protocoles de routage de l'information récoltée, pour la couverture de la zone d'intérêt, pour l'agrégation des données, etc. Elle est la première tâche exécutée par les nœuds après leur déploiement. La connaissance des positions des capteurs dans l'environnement surveillé est souvent

indispensable pour une grande majorité des applications (militaires, surveillances des patients, les établissements spécialisés (maison de retraite, EHPAD), détection des feux dans la forêt, surveillances des maisons intelligentes, suivis des animaux, .etc.), afin de pouvoir déterminer l'origine des événements détectés. « Où ? » est la question qui suit immédiatement la détection d'un événement (par exemple, où est le feu ?, aussi, se trouve le patient qui nécessite aide immédiatement ?). En outre, la localisation peut être utilisée dans les protocoles de routage géographique dans les réseaux à grande échelle, en transmettant les données seulement dans la direction de la destination [Hao et al., 2018].

### 3.4 CONCEPTION ET ÉVALUATION DE L'ALGORITHME DE LOCALISATION

Pour concevoir ou analyser l'applicabilité d'un algorithme de localisation à une application du RCSF donnée, il faut prendre en compte une série de facteurs, notamment les besoins en ressources de l'algorithme, la topologie du réseau, la nature du terrain dans lequel les RCSFs seront déployés et la densité des nœuds dans le réseau [Bachrach et Taylor, 2005]. Nous discutons de ces facteurs dans cette sous-section.

### 3.4.1 La densité des nœuds

La densité des nœuds dans le RCSF a une influence majeure sur les types d'algorithmes de localisation adaptées au réseau. Pour les algorithmes basés sur le nombre de sauts, par exemple, une densité élevée est nécessaire pour garantir la précision des distances approximatives [Rai et Daruwala, 2019]. Lorsque les nœuds de références font partie du processus de localisation, leur densité doit être suffisamment élevée pour que l'opération de localisation soit efficace. En général, de nombreux algorithmes de localisation exigent une certaine densité de nœuds, Un seuil en dessous de laquelle l'erreur de localisation peut augmenter de manière significative mais au-dessus de laquelle l'erreur ne diminue que très légèrement.

Par exemple, dans une simulation de localisation pilotée par des références basées sur 500 nœuds déployés dans une zone de 100 m 100 m 100 m [Yunfeng Han et al., 2015], on constate, pour un pourcentage d'ancrage de 20%, que la couverture de localisation a augmenté de près de 50% lorsque la densité des nœuds (représente le nombre attendu des nœuds dans le voisinage) est passée de 8 à 11, mais a peu augmenté lorsque la densité des nœuds est passée de 12 à 16.

### 3.4.2 Les facteurs environnementaux

Les obstacles tels que les bâtiments, les rochers et les arbres dans la zone où le RCSF est déployé peuvent entraver les signaux utilisés pour la mesure des gammes de signaux (méthodes de télémétrie discutées à la section 6.3.1.1) et entraîner un processus de localisation erratique. Par exemple, les signaux réfléchis par des obstacles physiques situés dans le RCSF peuvent interférer les uns avec les autres, entraînant des effets de trajets multiples et des erreurs de localisation associées [Amundson et Koutsoukos, 2009] [Wang et al., 2018]. Outre les obstacles physiques, d'autres facteurs environnementaux tels que les précipitations et la quantité d'humidité dans l'air sont bien connus pour affecter la propagation des ondes radio, ce qui peut entraîner des erreurs pour les techniques de localisation qui reposent sur les ondes radio [Kalyani, 2018]. Un bon algorithme de localisation devrait avoir des mécanismes pour se prémunir contre les erreurs causées par des facteurs environnementaux tels que ceux répertoriés ici ou pour s'en remettre.

### 3.4.3 La topologie du réseau

Les topologies des RCSFs irrégulières entraînent généralement des erreurs de localisation plus importantes [Mao et al., 2007a]. Même là où la topologie n'est pas irrégulière, les nœuds en bordure du RCSF ont généralement une tendance à être relativement difficiles à localiser car ils :

- 1- ils ont un petit nombre de voisins.
- 2- ils ont tous leurs voisins d'un côté, ce qui implique que les mesures de plage pour ces nœuds ne fournissent qu'une perspective limitée de leur emplacement.

Un bon algorithme de localisation devrait être capable de faire les compensations nécessaires pour les erreurs résultant des artefacts de topologie.

### 3.4.4 Les contraintes de ressources

La conception de l'algorithme de localisation doit tenir compte du fait que les nœuds du RCSF ont une puissance de traitement et une mémoire limitée. Pour les applications où la faible précision des mesures de localisation est adéquate, des algorithmes approximatifs qui peuvent estimer la position en utilisant une faible puissance et un bon matériel offrent une bonne option pour relever le défi des limitations de ressources [Reichenbach et al., 2007]. Lorsqu'une application nécessite des informations de localisation avec une grande précision, les algorithmes exacts nécessaires à cette fin consomment généralement plus d'énergie. Un bon algorithme dans un tel cas devrait répartir les tâches de localisation entre les différents nœuds par exemple, les tâches réparties entre la station de base, les nœuds de références (qui sont généralement plus puissants que le nœud typique) et le reste des nœuds.

## 3.5 TAXONOMIE DE LOCALISATION DES NŒUDS

L'algorithme de localisation du RCSF peut être classé en fonction de ses stratégies de conception et de mise en œuvre telles que la connectivité des nœuds, les informations de plage, les informations d'ancrage et le modèle de calcul. Les algorithmes de localisation peuvent être classés, selon plusieurs critères, reflétant la conception fondamentale et le choix de mise en œuvre. Ces différents critères forment une taxonomie raisonnable pour caractériser et évaluer les algorithmes de localisation des nœuds sont illustrés dans la figure 3.1.

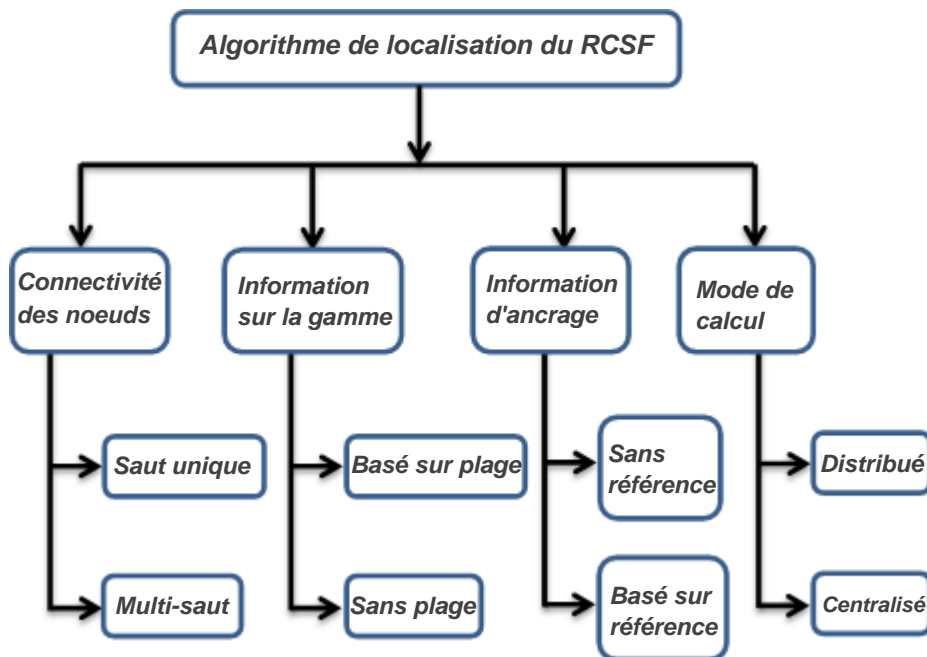


FIGURE 3.1 – Taxonomie des schémas de localisation pour les réseaux de capteurs.

Dans ces différentes alternatives de conception pour les algorithmes de localisation en général et dans les réseaux de capteurs sans fils en particulier. La localisation des nœuds est devenu un domaine de recherche le

plus actif du RCSF. La localisation des nœuds se réfère généralement au processus de détermination des positions d'un ou plusieurs nœuds dans un réseau des nœuds de grande quantités. De nombreuses applications proposées pour le RCSF nécessitent une connaissance de l'origine des informations de détection, ce qui pose un problème de localisation.

### 3.5.1 Algorithme à saut unique ou à saut multiples

Les algorithmes de localisation dans lesquels les nœuds cibles à localiser se trouvent dans un voisinage à un saut d'un nombre suffisant de nœuds d'ancrage sont appelés algorithmes à saut unique [Sharma et Saini, 2015]. Lorsque deux nœuds dans un RCSF sont séparés par une distance supérieure à la portée radio et que la densité des nœuds est suffisamment élevée pour créer un chemin continu entre eux, un tel chemin s'appelle chemin à sauts multiples. Dans de nombreuses applications des RCSFs telles que la surveillance d'un environnement étendu, il n'est pas toujours possible d'avoir des nœuds cibles dans le voisinage à un saut des nœuds d'ancrage [Minet et al., 2017]. Dans cette situation, le nœud s'appuie sur une localisation à sauts multiples comme indiqué ci-dessous dans la figure 3.2.

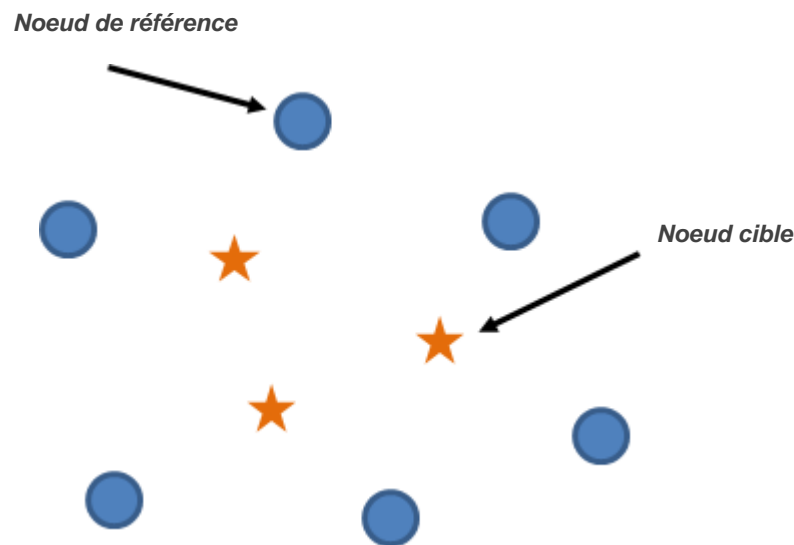


FIGURE 3.2 – Algorithme à saut multiple.

### 3.5.2 Algorithme sans ou basé sur la plage

Les paradigmes de localisation basés sur la plage de communication déploient un mécanisme de mesure complexe et dédié pour déduire des informations de la plage (en matière de distance d'estimation d'angle) pour calculer l'emplacement des nœuds cibles. Contrairement aux méthodes basées sur la plage, les schémas sans plage de communication ne nécessitent pas une gamme absolue d'informations pour l'estimation de l'emplacement. Étant donné que les métriques de classement indirect comme le nombre de sauts peuvent fournir une approximation grossière

seulement des distances euclidiennes entre les nœuds de capteurs, les algorithmes sans plage sont généralement moins précis que ceux basés sur la plage [Shahra et al., 2017].

### 3.5.3 Algorithme sans ou basé sur ancre

L'algorithme basé sur les ancres repose sur des nœuds (ancres ou balises) qui reçoivent leurs informations de position absolue soit par configuration manuelle, soit par un mécanisme de localisation dédié comme le GPS. Le but des algorithmes basés sur les ancres est de déterminer les coordonnées des nœuds cibles à l'aide des informations de position des nœuds d'ancrage. Dans l'autre côté, les algorithmes sans ancre sont utilisés pour trouver les emplacements relatifs des nœuds de capteurs à partir d'un ensemble de contraintes géométriques extraites des mesures de distance/proximité [Sobehy et al., 2018].

### 3.5.4 Algorithme centralisé ou distribué

Dans les paradigmes basés sur des algorithmes centralisés, les informations de distance entre l'ensemble des capteurs du réseau doivent être communiquées directement au processeur central qui s'appelle station de base, la station de base a pour tâche de calculer les coordonnées (détermination des positions des nœuds inconnus) du nœud cible est effectuée et les informations des coordonnées doivent être retransmises aux nœuds. Contrairement aux algorithmes centralisés, dans un schéma basé sur des algorithmes distribués ou décentralisés, chaque nœud (ou groupe de nœuds dans la même région) estime son emplacement à l'aide de mesures intercapteurs et d'informations de localisation collectées à partir des nœuds d'ancrage dans sa région de couverture du réseau [Zanjireh et Larijani, 2015].

## 3.6 SYSTÈME DE LOCALISATION

Les systèmes de localisation dans les réseaux de capteurs sans fils peuvent être divisés en trois phases distinctes [Boukerche et al., 2007] [Ahmad et al., 2017].

**L'estimation de distance / angle** dans cette phase chaque nœud référence travaille pour estimer la distance ou l'angle entre lui et les autres nœuds inconnus dans sa région de localisation. Ces informations enregistrées seront par la suite exploitées dans les étapes suivantes du système de localisation.

**le calcul de position** cette phase est responsable du calcul de la position d'un nœud sur la base des informations disponibles concernant les distances / angles et les positions des nœuds de référence, donc il utilise les informations enregistrées dans la phase d'estimation et fait appel aux techniques pour but de déterminer la position des nœuds cibles.

**l'algorithme de localisation** c'est la phase cruciale et pilier du système de localisation. Il a pour objectif de déterminer les positions exactes des nœuds cibles et comment les informations disponibles (sauvegardées dans les deux phases) seront manipulées afin de permettre au reste des nœuds

cibles du réseau de calculer ses positions. Les principales phases du système de localisation sont illustrées dans **la figure 3.3**. Aussi, à fournir un point de vue didactique, l'importance d'une telle division en phase dépend de la nécessité de reconnaître que la performance finale d'un système de localisation est relié directement de chacune de ces phases. En outre, chaque phase se caractérise par ses propres techniques de solutions et possède son propre objectif. Dans plusieurs travaux de recherche chaque phase est considérée comme des sous-domaines du problème de localisation [Ahmad et al., 2019] qui doivent être traités et étudiés indépendamment :

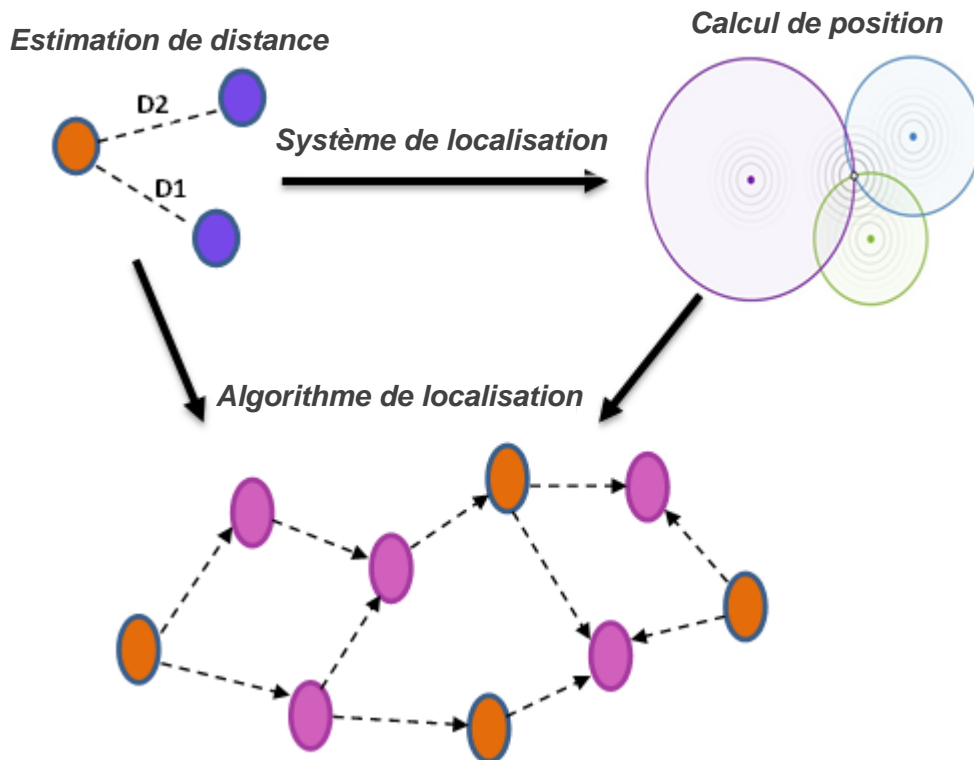


FIGURE 3.3 – Schéma de localisation.

### 3.6.1 Estimation distance / angle

L'estimation distance / angle consiste à estimer la distance ou l'angle entre deux nœuds. Il existe différentes méthodes peuvent être utilisées pour estimer ces informations. Certaines de ces méthodes sont très précises, mais avec des coûts plus élevés (en matière de matériel, énergie et ressources processeur), tandis que d'autres sont moins précis mais déjà disponibles sur la plupart des nœuds de capteurs. Les algorithmes d'estimation pour les RCSFs peuvent être classés en deux catégories :

- (i) Des algorithmes basés sur la plage de communication.
- (ii) Des algorithmes sans plage de communication.

### Estimation de distance basée sur rayon de communication

Il existe plusieurs techniques basées du rayon de communication pour estimer la distance. Sur la base des informations de plage, l'emplacement d'un nœud est déterminé. Certains des algorithmes de localisation basées sur la plage incluent :

- 1 - L'indication de la force du signal reçu (RSSI) .
- 2 - Angle d'arrivée (AoA).
- 3 - Heure d'arrivée (ToA).
- 4 - Décalage horaire d'arrivée (TDoA).

### L'indicateur de force du signal reçu (RSSI)

Le signal radio transmis possède une relation inversible avec la distance traversée, i.e. Le signal radio transmis s'atténue à mesure que la distance entre l'émetteur et le récepteur augmente. Avec l'augmentation de la distance, la force du signal radio diminue de façon exponentielle. L'atténuation de la force du signal est mesurée par le circuit d'indicateur de force du signal reçu par les récepteurs (RSSI). RSSI estime la distance parcourue par un signal au récepteur en mesurant la puissance du signal reçu. La diminution de la puissance transmise au niveau du récepteur peut être calculée et traduite en une distance estimée. Un modèle de propagation radio prédit la distance  $d$  par l'équation suivante :

$$P_r(d) = \frac{P_\lambda * G_t * G_r * \lambda^2}{4 * \pi^2 * d^n * L} \quad (1)$$

D'où  $P_\lambda$  représente la puissance transmise,  $G_t$  et  $G_r$  est le gain d'antenne de l'émetteur et du récepteur respectivement,  $L$  est la perte du système et  $\lambda$  est la longueur d'onde du système.  $G_t$ ,  $G_r$  et  $L$  sont définis à une valeur fixe égale à 1. L'utilisation de RSSI dans le calcul de la distance est reformulée dans [Zheng et al., 2011] et interprétée par l'équation suivante :

$$P_r(d) = P_r(d_0) + 10 * \eta * \log\left(\frac{d}{d_0}\right) + X_\sigma \quad (2)$$

où  $d$  est la distance de l'émetteur au récepteur,  $\eta$  est l'exposant de perte de trajet qui mesure le taux auquel le RSSI diminue avec la distance,  $X_\sigma$  est une variable aléatoire distribuée gaussienne à moyenne nulle dont la valeur moyenne est nulle et qui reflète le changement du signal reçu à une certaine distance,  $d_0$  est la distance de référence, généralement égale à un mètre,  $P_r(d_0)$  est la puissance calculée à une distance de référence  $d_0$  de l'émetteur. La plupart des outils qui fournissent la mesure RSSI montrent la relation entre la puissance d'émission et la puissance de réception par la formule présentée dans [Jenn, 2008] et elle est reformulée par l'équation suivante :

$$P_r = \frac{P_t}{d^\eta} \quad (3)$$

A partir de l'équation ci-dessus, nous obtenons,

$$P_r(dBm) = A - 10 * \eta * \log(d) \quad (4)$$

Où  $P_r$  est la puissance du signal reçu (RSSI),  $A$  est la puissance du signal à une distance d'un mètre. En utilisant l'équation ci-dessus, nous pouvons facilement calculer la distance. La précision de RSSI dépend du modèle de perte de chemin. En effet, RSSI est affecté par la décoloration rapide, la mobilité, les ombres, terrain. Des recherches ont été introduit dans [Savarese et al., 2002] ont rapporté que l'erreur de plage introduite par RSSI est de  $\pm 50\%$ . Cela peut être réduit en prenant la moyenne du nombre de mesures à une certaine distance. Le modèle d'observation est choisi pour indiquer l'atténuation de la puissance dans le processus de transmission du signal sans fil, en raison de divers obstacles et facteurs d'interférence, il existe différents types d'environnement de propagation du signal sans fil, il peut être reflété l'environnement de mesure réelle en déterminant les paramètres environnementaux associés dans le modèle.

### Evaluation de RSSI

Le principal avantage est son faible coût, car la plupart des récepteurs sont capables d'estimer la force du signal reçu.

L'inconvénient de cette méthode est qu'elle est très sensible au bruit et aux interférences, ce qui se traduit par des imprécisions plus élevées des estimations de distance. Certaines expériences [Savvides et andand M. B. Strivastava, 2001] montrent des erreurs de 2 à 3 m dans des scénarios où tous les nœuds sont placés dans un champ plan à 1,5 m du sol avec une portée de communication de 10 m. Bien que RSSI montre des résultats plausibles dans les simulations et les expériences contrôlées, son utilisation dans des applications du monde réel est encore discutable [He et al., 2003]. Mais, compte tenu de son faible coût, il est possible qu'une utilisation plus sophistiquée et précise du RSSI (par exemple avec de meilleurs émetteurs) devienne la technique d'estimation de distance la plus utilisée du point de vue coût / précision [Bachrach et Taylor, 2005]. Cependant, cette technique n'est pas encore disponible.

### Temps d'arrivée (Time of Arrival)

Cette technique estime la distance en calculant le temps requis par un signal pour traverser de l'émetteur au récepteur. Les types de signaux utilisés sont les suivants : RF, acoustique, infrarouge et ultrasons. Les appareils compatibles avec GPS utilisent cette technique pour estimer la distance. L'estimation de la distance à l'aide de ToA est illustrée dans la figure 3.4.

Soit le nœud A l'émetteur et B le récepteur,  $t_a$  est l'instant auquel un signal est transmis depuis A et  $t_b$  le moment où il est reçu en B,  $v$  représente la vitesse d'un signal. Alors la distance entre les nœuds A et B sont estimés par l'équation suivante :

$$d = (t_a - t_b) * v \quad (5)$$

Comme les nœuds ne sont pas généralement synchronisés, la distance entre les nœuds à diverses instances, telle que calculée ci-dessus, peut varier. De plus, la vitesse du signal (principalement des signaux ultrasons) peut varier. En effet, ils sont souvent affectés par les facteurs de la nature : la température, l'humidité et la pression, par conséquent, pour

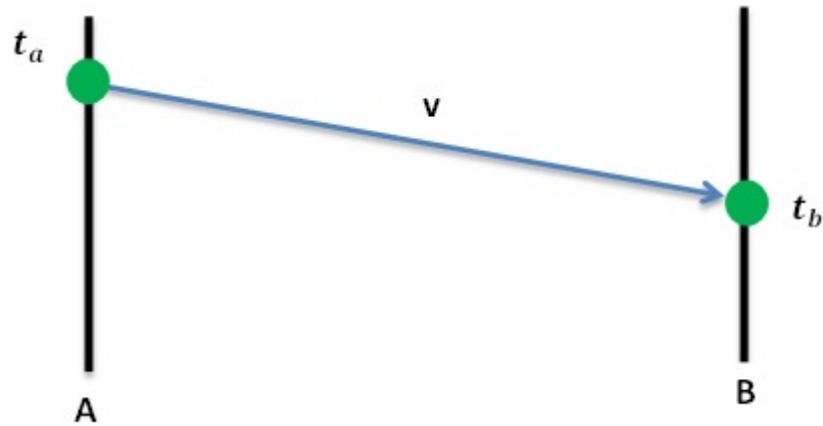


FIGURE 3.4 – Estimation de la distance à l'aide de ToA.

supprimer ou surmonter le problème de synchronisation, ToA est formulé par un système appelé temps aller-retour (round trip time : RTT). Le modèle de RTT est présenté dans la figure 3.5. Le nœud A transmet le signal à l'instant  $t_{a1}$  et le nœud B reçoit à l'instant  $t_{b1}$

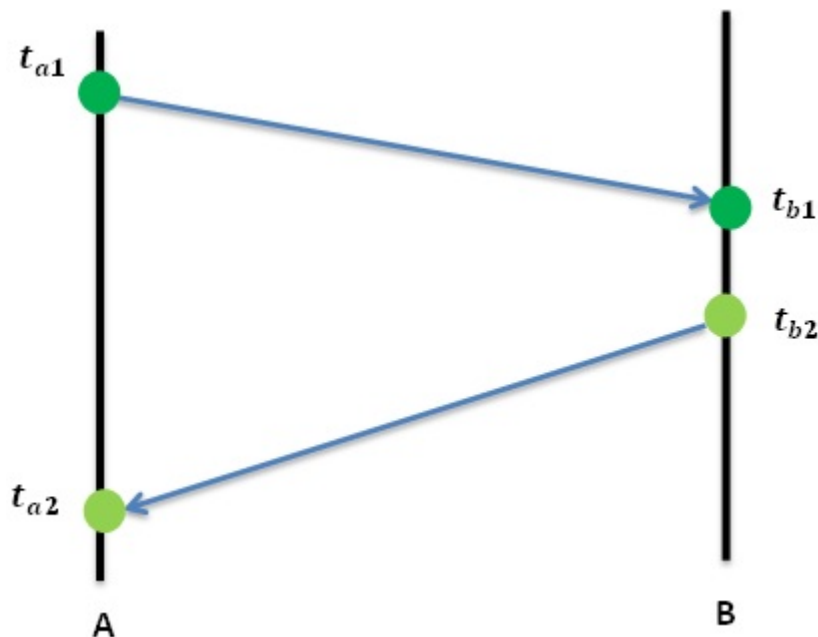


FIGURE 3.5 – Estimation de la distance de ToA basée sur RTT.

Après un certain traitement, le nœud B retransmet le signal au nœud A à l'instant  $t_{b2}$  et le nœud A le reçoit à l'instant  $t_{a2}$ . Finalement, la formule de calcul de la distance  $d$  s'écrit par l'équation suivante :

$$d = \frac{(t_{a1} - t_{a2}) * (t_{b2} - t_{b1}) * v}{2} \quad (6)$$

#### Evaluation de ToA :

On suppose que le chemin parcouru par le signal est symétrique. Cependant, ToA fournit un bon niveau de précision, mais nécessite un trai-

tement relativement rapide des nœuds de capteurs afin de résoudre les différences de synchronisation pour une mesure de distance précise. En outre, la précision de ToA dépend de la capacité du récepteur à estimer avec précision l'heure d'arrivée du signal reçu. Ceci est souvent affecté par le signal de trajets multiples et l'ombrage.

### Différence du temps d'arrivée

La différence du temps d'arrivée ( TDoA :Time-Difference of Arrival ) est basée sur la même approche utilisée par ToA. Mais cette fois-ci, il utilise deux signaux différents, par exemple radiofréquence RF et un signal ultrasonore de vitesse différente. Et donc cela permet de supprimer le besoin de synchronisation entre les nœuds. Dans TDoA, chaque nœud est équipé d'un haut-parleur et d'un microphone. Plusieurs travaux de recherche ont utilisé TDoA pour l'estimation de la distance pour réaliser des systèmes de localisation [Priyantha et al., 2000] [Ward et al., 1997] [Priyantha et al., 2001] , L'estimation de la distance à l'aide du TDoA est illustrée dans la figure 3.6. Le nœud A transmet un signal radio avec la vitesse  $v_1$  à l'instant  $t_{a1}$  et dans l'autre côté le nœud B reçoit le signal à l'instant  $t_{b1}$ . La formule de la distance  $d$  est calculée par l'équation suivante :

$$d = (t_{b1} - t_{a1}) * v_1 \quad (7)$$

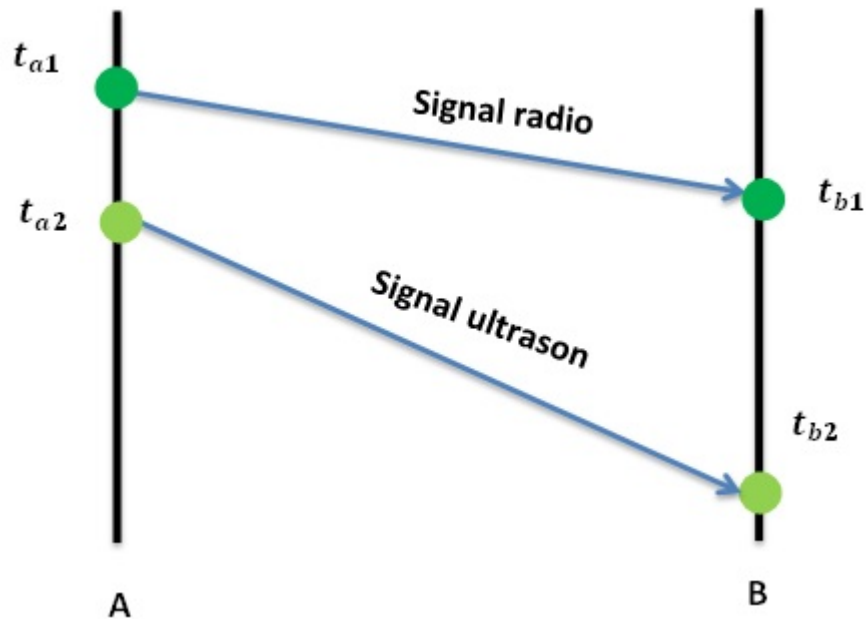


FIGURE 3.6 – Estimation de distance basée sur TDOA.

Après un certain retard , le nœud A transmet le signal ultrasonore via la vitesse  $v_2$  à l'instant  $t_{a2}$  et le nœud B reçoit le signal à l'instant  $t_{b2}$ . La valeur de la distance  $d$  est calculée par l'équation suivante :

$$d = (t_{b2} - t_{a2}) * v_2 \quad (8)$$

En résolvant les deux équations 7 et 8, nous obtenons la formule finale de la distance  $d$  résumé par l'équation suivante :

$$d = ((t_{b2} - t_{a2}) - (t_{b1} - t_{a1})) * \left[ \frac{v_1 * v_2}{v_1 - v_2} \right] \quad (9)$$

### Evaluation de TDOA

TDOA fonctionne efficacement dans des conditions de visibilité directe. Mais il est difficile de parvenir à une condition de visibilité directe dans certains environnements besoin du matériel supplémentaire tel que haut-parleurs, microphones. En outre, il supprime le besoin de synchronisation. Les haut-parleurs et les microphones utilisés doivent être correctement calibrés et les signaux ne doivent pas être affectés par des facteurs externes comme dans ToA.

### Angle/Direction d'arrivée

Le concept de base de l'estimation d'emplacement basé sur l'angle d'arrivée (AoA : Angle of Arrival) se fait en utilisant des antennes directives ou un réseau de récepteurs généralement trois ou plus qui sont uniformément séparés. En premier lieu, les nœuds de référence transmettent des signaux à l'aide d'antennes omnidirectionnelles [Voigt et al., 2013], le nœud cible reçoit les signaux des nœuds d'ancrage voisins et donc peut mesurer le signal reçu AoA. Si le nœud cible connaît sa propre orientation, seuls deux nœuds d'ancrage sont nécessaires pour déterminer l'emplacement du nœud cible. Un nœud connaît son orientation s'il connaît la direction du Nord ou une direction communément connue des nœuds de référence et du nœud cible.

### Evaluation d'AOA

(i) Le coût du matériel : chaque nœud doit avoir un haut-parleur et plusieurs microphones/réseau d'antennes, cela augmente le coût ainsi que la taille du nœud.

(ii) l'approche n'évolue pas bien pour les réseaux avec un plus grand nombre de nœuds.

(iii) Besoin d'une horloge à très haute résolution pour produire le résultat d'une précision acceptable.

Le besoin de matériel supplémentaire et d'une distance minimale entre les récepteurs entraîne certains inconvénients en matière de coût et de taille des nœuds.

Une comparaison qualitative de ces techniques d'estimations basées sur les plages est présentée dans la table 3.1.

### Estimation de distance sans rayon de communication

Les algorithmes de localisation sans rayon utilisent des informations de connectivité entre un nœud inconnu et des points de repère. Un point de repère peut obtenir ses informations de localisation à l'aide du GPS ou par le biais d'une information déployée artificiellement. Certains des algorithmes de localisation sans plage incluent :

TABLE 3.1 – Comparaison qualitative des techniques de localisation basées sur l'aire de répartition

Technique	Matériel	Problème	Précision
RSSI	Aucun	Interférence	(2 - 3 m)
ToA	Aucun	Synchronisation	(2 - 5 cm)
TDoA	Hautparleur, microphones	—	(2 - 5 cm)
AoA	Réseaux de microphones	Directivité et Ombrage	Quelques degrés

### Nombre de sauts

Les nœuds capteurs sont déployés de manière à ce que chaque nœud reste dans la plage de ses nœuds voisins, c'est-à-dire qu'un nœud se trouve dans la plage  $R$  de son nœud voisin. Connaissant le nombre de sauts (nombre de sauts) et la longueur d'un saut (longueur de saut), la distance  $d$  entre deux nœuds est calculée comme :

$$d = (\text{nombre des sauts}) * (\text{longueur du saut}) \quad (10)$$

### Evaluation

Cependant, la métrique du nombre de sauts a une certaine limitation :

- (i) Les nœuds ne formant pas une forme convexe peuvent ne pas trouver le nombre de sauts précis, cela est dû aux obstacles se trouvant sur le chemin le plus court vers le voisin.
- (ii) La mesure de la distance est toujours multiple de la longueur du saut, par ailleurs, il y a d'autres approches d'estimation de distance [Wong et al., 2005] [Tan et al., 2020].

## 3.6.2 Calcul de position

### Tri-lateration /Multilateration

La tri-lateration est une technique géométrique utilisée pour déterminer l'emplacement d'un nœud inconnu à l'aide de trois nœuds de références. Il utilise la distance entre les nœuds de références et le nœud inconnu. Un schéma de cette technique géométrique pour la localisation d'un nœud inconnu  $(x_u, y_u)$  avec des nœuds d'ancrage  $(x_i, y_i)$  est illustré dans la figure 3.7. La mesure de la distance basée sur plus de trois nœuds ancres est connue sous le nom de Multilateration.

Cette technique peut être utilisée pour obtenir un emplacement unique. Nous illustrons la Multilateration dans un espace de 2 dimensions avec des distances connues entre les nœuds d'ancrage et un nœud inconnu, les distances sont présentées comme suit :

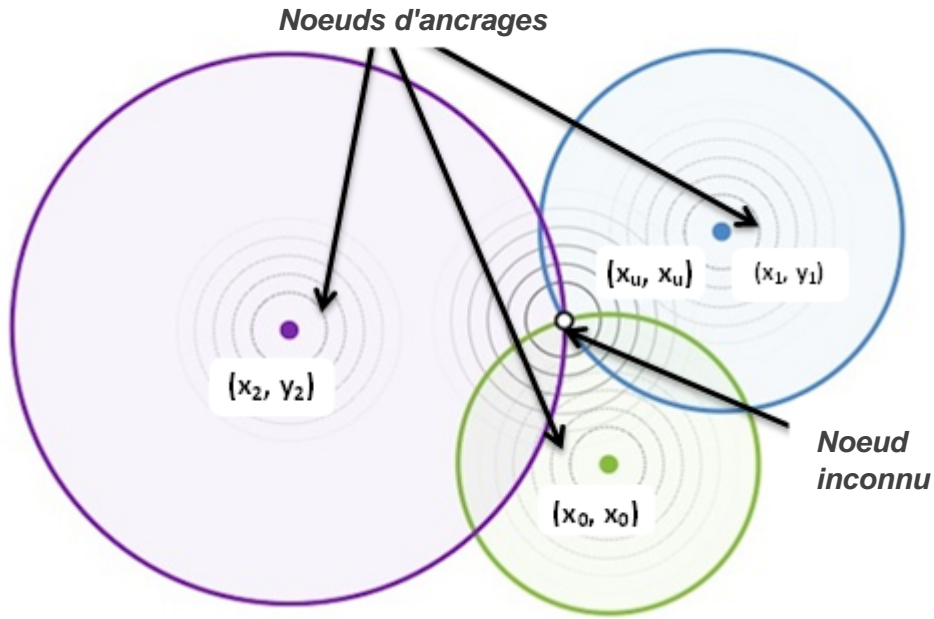


FIGURE 3.7 – Tri-lateration et Multilateration.

$$d_1^2 = (x_1 - x_u)^2 + (y_1 - y_u)^2 \quad (11)$$

$$d_2^2 = (x_2 - x_u)^2 + (y_2 - y_u)^2 \quad (12)$$

⋮

$$d_n^2 = (x_n - x_u)^2 + (y_n - y_u)^2 \quad (13)$$

La soustraction de l'équation (11) de (12) ... (13) donne :

$$d_2^2 - d_1^2 = x_2^2 - x_1^2 - 2 * (x_2 - x_1) * x_u + y_2^2 - y_1^2 - 2 * (y_2 - y_1) * y_u \quad (14)$$

$$d_3^2 - d_1^2 = x_3^2 - x_1^2 - 2 * (x_3 - x_1) * x_u + y_3^2 - y_1^2 - 2 * (y_3 - y_1) * y_u \quad (15)$$

⋮

$$d_n^2 - d_1^2 = x_n^2 - x_1^2 - 2 * (x_n - x_1) * x_u + y_n^2 - y_1^2 - 2 * (y_n - y_1) * y_u \quad (16)$$

Ré-arrangement, (14) .. (16) sous forme matricielle, on obtient :

$$\begin{bmatrix} x_2 - x_1 & y_2 - y_1 \\ x_3 - x_1 & y_3 - y_1 \\ \vdots & \vdots \\ x_n - x_1 & y_n - y_1 \end{bmatrix} \begin{bmatrix} x_u \\ y_u \end{bmatrix} = \frac{1}{2} \begin{bmatrix} x_2^2 + y_2^2 - d_2^2 - (x_1^2 + y_1^2 - d_1^2) \\ x_3^2 + y_3^2 - d_3^2 - (x_1^2 + y_1^2 - d_1^2) \\ \vdots \\ x_n^2 + y_n^2 - d_n^2 - (x_1^2 + y_1^2 - d_1^2) \end{bmatrix} \quad (17)$$

La matrice ci-dessus peut être réécrite comme suit :

$$Au = b \quad (18)$$

$$\text{Où : } A = \begin{bmatrix} x_2 - x_1 & y_2 - y_1 \\ x_3 - x_1 & y_3 - y_1 \\ \vdots & \vdots \\ x_n - x_1 & y_n - y_1 \end{bmatrix}, u = \begin{bmatrix} x_u \\ y_u \end{bmatrix},$$

$$b = \begin{bmatrix} x_2^2 + y_2^2 - d_2^2 - (x_1^2 + y_1^2 - d_1^2) \\ x_3^2 + y_3^2 - d_3^2 - (x_1^2 + y_1^2 - d_1^2) \\ \vdots \\ x_n^2 + y_n^2 - d_n^2 - (x_1^2 + y_1^2 - d_1^2) \end{bmatrix}$$

Par conséquent,  $u$  peut être dérivé comme :

$$u = (A^T.A)^{-1} * A^T.b \quad (19)$$

### Triangulation

La triangulation est une technique géométrique qui utilise l'angle d'arrivée pour déterminer l'emplacement des capteurs [Miu et al., 2001] [Niculescu et Badri Nath, 2003], et donc les informations sur les angles sont utilisées à la place des distances. Le calcul de la triangulation [Wei Ye et al., 2002] est illustré dans **la figure 3.8**, le nœud inconnu estime son angle par rapport à chacun des trois nœuds de référence et par la suite sur la base de ces angles et des positions des nœuds de références (qui forment un triangle), calcule sa propre position en utilisant relations trigonométriques simples des sinus et cosinus. Cette technique est similaire à la technique de tri-lateration. En fait, triangulation est basée sur la technique d'angle d'arrivée (AoA).

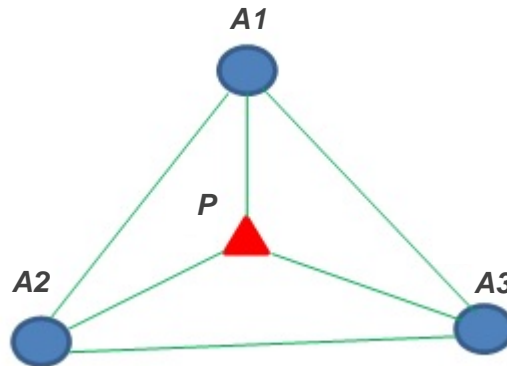


FIGURE 3.8 – *Triangulation.*

### 3.6.3 Algorithme de localisation

### 3.7 ALGORITHME DE LOCALISATION (ÉTAT DE L'ART)

La capacité d'auto-localisation est hautement souhaitable dans les applications de surveillance environnementale telle que la détection d'intrusion, la surveillance du trafic routier, la surveillance des patients à distance, etc.

la solution intuitive est que chaque nœud de capteur peut être localisé à

l'aide du système de navigation par un système de positionnement global (GPS). Cependant, des considérations pratiques telles que le coût, la consommation d'énergie et le volume des récepteurs GPS rendent impossible l'utilisation du GPS sur chaque nœud de capteur, en particulier pour le RCSF à large plage, par exemple, les nœuds de capteurs sont généralement alimentés par de petites batteries, qui ne peuvent généralement pas être facilement changées ou rechargées, de plus, le RCSF est généralement déployé à l'intérieur où la transmission des signaux satellites est bloquée par les obstacles.

Un exemple d'illustration est présenté dans la figure 3.9, dans un champ de captage bidimensionnel, on suppose que les coordonnées du nœud inconnu sont  $(x, y)$  qui doivent être positionnés ou localiser, les coordonnées du nœud de référence sont  $(x_1, y_1)$ ,  $(x_2, y_2)$ ,  $(x_3, y_3)$ . Le nœud inconnu sera localisable s'il y a au moins trois ancres.

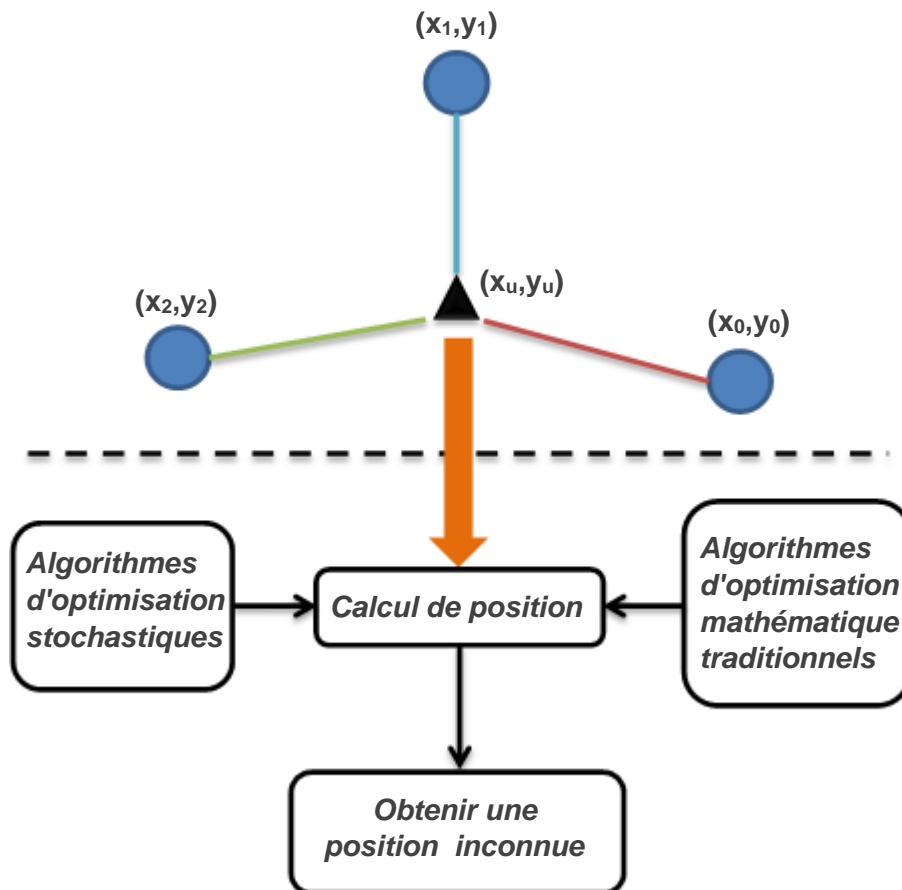


FIGURE 3.9 – Processus de localisation dans RCSF.

L'algorithme de localisation a reçu énormément d'attention au cours de la dernière décennie et plusieurs travaux de recherche ont été adressés pour résoudre le problème de localisation, en premier lieu, les approches mathématiques sont proposées pour traiter les problèmes de localisation des nœuds.

### 3.7.1 Les approches mathématiques

Une optimisation convexe basée sur une programmation semi-définie [Doherty et al., 2001] est proposée pour résoudre le problème de localisation des nœuds. La connectivité du réseau était représentée comme un ensemble de contraintes de localisation convexe pour le problème d'optimisation. L'idée de base c'est que les contraintes de proximité des nœuds qui sont à «distance d'audition» les uns des autres sont modélisées comme des contraintes convexes. Ensuite, un problème de faisabilité peut être résolu par des techniques de programmation convexe efficaces tel que programmation linéaire, programmation semi-définie et la méthode de cône de seconds ordres.

Une méthode de relaxation en programmation semi-définie (SDP) basée sur [Doherty et al., 2001] est proposée pour résoudre le problème de localisation en utilisant des informations de distance incomplètes et inexacts [Biswas et al., 2006]. Le problème est amélioré pour trouver un ensemble de positions de capteurs telles que les contraintes de distance données sont satisfaites. Les contraintes non convexes dans la formulation sont ensuite relâchées afin de produire un programme semi-défini qui peut être résolu efficacement. Le modèle de base est étendu afin de tenir compte des informations de distance bruyantes. En particulier, une méthode SDP distribuée itérative est proposée pour résoudre des programmes semi-définis à très grande échelle qui découle de problèmes de localisation pour les grands réseaux et sont intraitables en utilisant des méthodes centralisées. Le calcul de performance est évalué en basant sur deux métriques : la précision d'estimation et de temps de calcul.

Une classe de relaxation convexe basée sur une formulation de maximum de vraisemblance (MV) est proposée pour résoudre le problème de localisation des nœuds [Simonetto et Leus, 2014], cette classe a dérivé une version basée sur les bords efficaces de calcul de cette classe de relaxation convexes de maximum de vraisemblance et conçu un algorithme distribué qui permet aux nœuds de capteurs de résoudre ces convexes basés sur les bords programmes locaux en communiquant uniquement avec leurs voisins proches. En utilisant cette version de relaxations convexes pour envoyer un message à l'original. La formulation non convexe peut offrir une poignée puissante sur le calcul de solutions précis.

#### Evaluation

Bien que ces techniques traditionnelles de recherche opérationnelle aient prouvé leur performance et leur importance pour résoudre le problème de localisation, mais dans l'autre côté, ces méthodes nécessitent énormes efforts de calcul et des frais généraux de communication ce qui augmente de façon exponentielle à mesure que la taille du problème augmente. Aussi, la capacité d'un RCSF de gérer des nœuds de capteurs de travail en croissance n'est pas garantie, en outre, le temps de localisation sera très important, ce qui influencera automatiquement sur la consommation d'énergie des nœuds et systématiquement sur la durée de vie du réseau.

### 3.7.2 Les approches métaheuristiques

Ces dernières années, les algorithmes méta-heuristiques jouent progressivement un rôle important dans l'optimisation de l'ingénierie [Simon, 2008] [Blum et Roli, 2003b]. Cela est attribué au fait qu'ils nécessitent une mémoire et des efforts de calcul modéré et plus important encore, ils peuvent fournir de résultats très satisfaisants par rapport aux méthodes d'optimisation mathématiques traditionnelles. Le challenge de la localisation des nœuds dans les réseaux de capteurs sans fils est souvent formulé comme un problème d'optimisation multidimensionnelle.

Une étude générale de la localisation des nœuds basée sur les approches bio-inspirées est présentée dans [Kulkarni et al., 2009] [Kumar et al., 2012]. Pour répondre aux exigences de localisation du réseau de capteurs sans fil à grande échelle; L'approche de recuit simulé (RS) est proposée comme un algorithme de localisation [Kannan et al., 2006] basé sur deux phases, la première phase obtient une estimation précise de l'emplacement, et en deuxième phase : si certains nœuds ont un problème d'ambiguïté de retournement, l'optimisation est effectuée sur la base des informations de voisinage des nœuds et déplacée vers la position correcte.

#### Les techniques centralisées :

L'algorithme d'optimisation de l'essaim particulier (PSO : Particle swarm optimization) est une technique d'optimisation stochastique robuste basée sur le mouvement et l'intelligence des essaims. L'algorithme PSO est proposé comme solution adressée au problème de localisation [Gopakumar et Jacob, 2008], en se basant sur un nombre minimal des nœuds ancrés on peut localiser le reste des nœuds inconnus en utilisant l'algorithme PSO.

Étant donné,  $M$  nœuds sont déployés aléatoirement dans un réseau de captage de deux dimensions, dont  $K$  ( $K=M-N$ ) représente les nœuds de références déployés pour localiser les  $N$  nœuds inconnus. L'objectif est d'estimer les coordonnées  $(x, y)$  de  $N$  nœuds.

Donc pour un problème de localisation de deux dimensions, un total de  $2-N$  coordonnées non identifiées,  $\theta = [\theta_x, \theta_y]$

où :

$$\begin{aligned}\theta_x &= [x_1, x_2, \dots, x_n] \\ \theta_y &= [y_1, y_2, \dots, y_n]\end{aligned}$$

Doivent être estimées à l'aide des coordonnées du nœud d'ancrage :

$$[x_{n+1}, \dots, x_{n+m}] \text{ et } [y_{n+1}, \dots, y_{n+m}]$$

Si  $(x, y)$  sont les coordonnées du nœud cible à déterminer, alors la distance entre le nœud cible et le  $i$ ème nœud d'ancrage  $d_i$  sera :

$$d_i = \sqrt{(x - x_i)^2 + (y - y_i)^2} \quad (20)$$

L'idée de base de cette approche est que tous les nœuds envoient leurs coordonnées calculées et la distance estimée à la station de base, par la suite la station de base exécute l'algorithme de PSO pour minimiser l'erreur de localisation définie comme fonction objective, la formule de la fonction objective est définie comme suit :

$$f(x, y) = \frac{1}{M} \sum_{i=1}^M (\sqrt{(x - x_i)^2 + (y - y_i)^2} - \hat{d}_i)^2 \quad (21)$$

Où  $(x, y)$  est l'emplacement du nœud qui doit être déterminé et  $(x_i, y_i)$  sont les coordonnées du nième nœud de référence.  $M \geq 3$  est le nombre de nœuds de références dans la plage de transmission du nœud cible. En effet, la distance  $\hat{d}_i$  est défini par la formule suivante :

$$\hat{d}_i = [d_i + n_i] \quad (22)$$

Et  $n_i$  La variance du bruit affecte la précision de localisation.

### Les avantages :

- 1- La méthode fonctionne bien si les références possèdent une portée suffisante pour couvrir la région des nœuds inconnue .
- 2- l'erreur de localisation obtenue est la moitié par rapport à l'algorithme de recuit simulé.
- 3- Le PSO surmonte le problème des minima locaux généré par l'algorithme de recuit simulé.

Cette approche est exécutée sur une architecture centralisée pour les RCSFs, dont lequel, les mesures de distance des nœuds de référence voisines sont communiquées directement avec la station de base centrale.

### Cette technique présente des limites majeures récapitulait comme suit :

- Le système ne fonctionne bien que lorsque le nœud cible est entouré par trois nœuds références ou plus.
- Une estimation de la portée de tous les nœuds cibles situés autour des nœuds ancrés doit être effectuée par la station de base, ce type d'architecture nécessite beaucoup de communication qui peut conduire à des congestions, des retards, mais aussi un épuisement énergétique.
- Le schéma proposé à une évolutivité limitée car la dimensionnalité PSO est le double du nombre des nœuds cibles.
- Le problème d'ambiguïté de retournement [Guo et al., 2019] [Liu et al., 2016] est présenté dans la figure 3.10, alors que ce problème n'est pas traité dans cette approche, cela influence sur l'erreur de localisation et sur le taux de localisation.
- L'approche proposée ne fonctionne que dans le modèle centralisé de RCSF, alors que ce modèle nécessite une large gamme de communication.

En 2011, l'algorithme d'optimisation des abeilles (BOA : Bee Optimization Algorithm) est proposé pour la localisation des nœuds [Moussa et El-Sheimy, 2010] , où TOA et RSSI sont utilisés dans la phase d'estimation, d'autre part, deux modèles de déploiement sont exécutés dans un réseau de simulation, le premier est le meilleur déploiement dans lequel les nœuds cibles sont entourés de quatre nœuds de balises (placés dans le coin) avec des directions différentes, le second est un autre environnement dont lequel les nœuds de balises sont déployés dans le milieu orienté dans une seule direction par rapport à un nœud inconnu, le résultat de simulation obtenu prouve que l'algorithme BOA est efficace et a des performances remarquables, en outre une étude comparative est menée entre l'approche proposée et Cramer-Rao Bound (CRB), le résultat final a démontré que le taux de précision de l'approche proposée basée

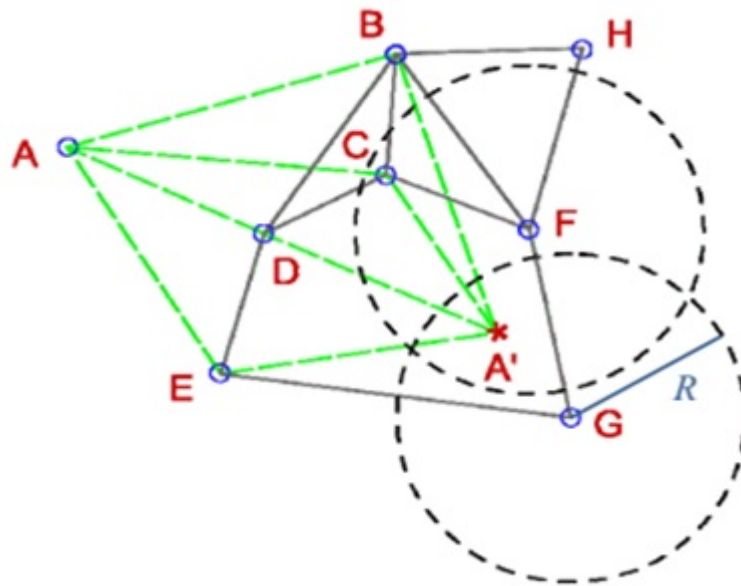


FIGURE 3.10 – Phénomène d'ambiguïté.

sur TOA est meilleur que la mesure RSSI. Cependant, certaines critiques doivent être signalées :

- BOA est adapté au défi de localisation des nœuds mais il n'y a pas d'étude comparative avec les autres approches méta-heuristiques.
- Le défi de l'évolutivité n'est pas abordé dans cette approche.
- Les paramètres BOA ne sont pas évalués.
- La présente étude n'a étudié que les erreurs de localisation, alors que le temps de localisation et la consommation d'énergie sont des facteurs décisifs pour l'efficacité de l'approche proposée.

**Une technique de la localisation itérative** est proposée basant sur l'algorithme de PSO, cette approche est exécutée dans un environnement distribué [Kulkarni et al., 2009]. Un nœud inconnu qui est entouré de trois nœuds de référence ou plus dans sa plage de communication exécute l'algorithme PSO pour minimiser l'erreur de localisation qui a utilisé la même fonction que le travail [Gopakumar et Jacob, 2008].

Les nœuds localisés à l'itération actuelle servent comme des nœuds de références dans la prochaine itération. Ce processus se poursuit de manière itérative, jusqu'à ce que tous les nœuds soient localisés ou qu'aucun autre nœud ne puisse être localisé. Cette méthode ne nécessite pas que chaque nœud transmette sa mesure de distance à un nœud central ou à la station de base. De plus, il peut localiser tous les nœuds qui ont trois nœuds de références dans leur plage de communication. Cependant, la localisation basée sur l'algorithme PSO nécessite des calculs complexes qui entraînent une puissance de calcul relativement important et un temps de calcul plus long. Une approche hybride de l'algorithme de recherche de nourriture bactérienne (BFA : bacterial Foraging aAlgorithm ) et l'algorithme de PSO est présentée dans [Abd-Elazim et Ali, 2013]. Il se concentre sur la localisation itérative de nœuds distribués basée sur la plage.

Le travail compare l'approche hybride avec l'algorithme PSO standard.

Il serait plus efficace en terme de capacité de recherche. D'un autre côté, l'algorithme de recherche de nourriture bactérienne serait moins gourmand en mémoire et plus précis. Une étude comparative des performances entre PSO et BFA est adressée en termes du nombre de nœuds localisés, de précision de localisation et de temps de calcul, il a été noté que les deux approches ont un problème commercial où les simulations obtenues ont montré que le PSO converge rapidement par contribution à BFA, par contre BFA donne de nombreuses précisions d'erreur que PSO.

#### **Les avantages :**

Les deux méthodes proposées qui n'exigent pas que chaque nœud transmette sa mesure de plage à un nœud central. En outre, il peut localiser tous les nœuds qui ont trois nœuds ancrés au minimum dans leur plage. Au fur et à mesure que les itérations de localisation passent, un nœud peut obtenir un plus grand nombre de références pour la localisation, ce qui atténue le problème d'ambiguïté de basculement, situation qui entraîne une grande erreur de localisation lorsque les références sont presque colinéaires. Cependant, la méthode proposée est sujette à l'accumulation d'erreurs [Qu, 2017].

**L'algorithme de chauve-souris** est un algorithme de recherche méta-heuristique inspiré des capacités fascinantes des chauves-souris telles que la recherche de leurs proies et la discrimination de différents types d'insectes même dans l'obscurité totale. La capacité d'écholocation avancée des chauves-souris les rend fascinantes. Les chauves-souris utilisent un sonar typique appelé écholocation pour détecter les proies et éviter les obstacles. Les chauves-souris sont capables de reconnaître les positions des objets en émettant des signaux audio élevés et courts et par collision et réflexion de ces signaux émis, l'algorithme de chauve-souris (Bat Algorithm) est proposé pour résoudre le problème de localisation des nœuds [Goyal et Patterh, 2013], l'approche proposée a prouvé son efficacité dont lequel un paradigme décentralisé est adopté via cette approche, en outre, les résultats obtenus montrent que le taux de localisation par l'algorithme proposé est très important et supérieur quand on le comparant avec BBO. L'algorithme de chauve-souris a démontré sa capacité et son adaptabilité pour le problème de localisation des nœuds. Bien que l'algorithme de chauve-souris soit puissant dans la phase de l'exploitation mais à une certaine insuffisance dans la phase d'exploration, l'algorithme proposé souffre de certains facteurs négatifs concernant ;

- 1- Les chauves-souris restent en phase d'exploration et ne peuvent pas explorer toutes les chauves-souris dans la zone de recherche.
- 2- Les paramètres de l'algorithme ne sont pas évalués.
- 3- Le challenge d'évolutivité n'est pas développé, seuls 40 nœuds sont déployés dans une zone de 100 mètres carrés, donc reste à prouver le développement durable de l'approche pour évaluer les performances de l'algorithme.
- 4- Deux critères (nombre de nœuds localisés et erreur de localisation) sont évalués, par ailleurs, le temps de localisation est négligé, alors que le temps de localisation joue un rôle décisif dans ce domaine. Et qui influence la durée de vie des nœuds et donc sur la résistance du réseau,

donc nous ne pouvons donc pas justifier que l'algorithme Bat soit meilleur que BBO si BBO est plus rapide.

5- Les paramètres sélectionnés de l'algorithme de chauve-souris ne sont pas régularisés au problème où la fréquence du pouls est très supérieure et la fréquence maximale est très inférieure.

**Algorithme à évolution différentielle (DE :differential evolution)** est un autre métaheuristique proposée pour déterminer les coordonnées des nœuds inconnus en se basant sur quelques nœuds de références [Harikrishnan et al., 2014], l'algorithme est implémenté dans un environnement distribué, d'où chaque nœud du réseau exécute l'algorithme d'indépendamment pour déterminer sa position, afin de minimiser l'erreur de localisation entre la position estimée et la position actuelle, l'approche a prouvé une convergence rapide et cela après quelques itérations (22 itérations), ce qui considère comme facteur très motivant vu le temps de localisation est minimale et donc permet de conserver l'énergie des capteurs et globalement prolonger la durée de vie du réseau, cependant, il y a quelques critiques à signaler :

- DE est facile à adapter pour le schéma de localisation des nœuds, mais aucune étude comparative n'est abordée dans cet article.
- Le problème d'évolutivité n'est pas traité pour tester le développement d'échelle de l'algorithme proposé.
- La présente étude n'a étudié que les erreurs de localisation, alors que le temps de localisation et la consommation d'énergie sont des facteurs cruciaux pour l'efficacité de l'approche proposée.

**Algorithme de Pollinisation des Fleurs (FPA : Flower Pollination Algorithm)** est un algorithme méta-heuristique bio-inspirées , cette approche est proposée pour traiter le challenge de localisation des nœuds , proposé dans [Goyal et Patterh, 2015] , l'idée fondamentale et d'évaluer les coordonnées des meilleurs nœuds en rapprochant ses voisins d'un bond, une étude comparative est abordée dans cette approche, les résultats de simulation obtenus (les coordonnées des nœuds calculés) par FPA ont été comparés aux différentes améliorations de l'algorithme PSO. De plus, plusieurs topologies de réseau sont exercées, les résultats de simulation montrent que le taux de localisation obtenu par l'algorithme FPA est supérieur aux différentes améliorations de l'algorithme PSO. L'approche proposée présente de nombreux avantages, ce qui n'empêche pas de mentionner des remarques cruciales :

- FPA possède plusieurs règles , paramètres à adapter et aussi des étapes à suivre , ce qui alourdit vraiment le déroulement et l'évolution de l'algorithme de telle sorte que le critère de temps de localisation influencera négativement sur la totalité des nœuds , d'où le temps de localisation se considère comme facteur important dans la localisation des nœuds et principalement par exemple dans les feux de forêt. [Aslan et al., 2012] , [Trivedi et Srivastava, 2014]

**L'algorithme de libellule (DA : Dragonfly Algorithm)** c'est un algorithme bio-inspiré, proposé comme solution adressée au problème de la localisation des nœuds dans [Daely et Shin, 2016] , un nombre maximale de 10 nœuds inconnus sont déployés dans le réseau carré où les

nœuds ancrés sont placés dans le coin (quatre coins) , plusieurs paramètres sont simulés pour prouver le développement et la résistance de l'algorithme proposé. Une étude comparative est menée avec l'algorithme PSO, le résultat des simulations a montré que l'erreur de localisation obtenue par l'algorithme DA est inférieure à PSO, des remarques importantes sont mentionnées pour cet algorithme :

- Le nombre de nœuds (14) est faible pour juger l'évolution de l'algorithme proposé (le défi de l'évolution n'est pas traité).
- L'approche proposée n'a étudié que l'erreur de localisation, alors que le temps de localisation et la consommation d'énergie sont les principaux facteurs de performance de l'approche proposée.

Une **étude comparative** est introduit entre les trois méta-heuristiques les plus populaires (Bat Algorithm, PSO et DE) dans [Mihoubi et al., 2017] , les performances d'évaluations sont articulées sur deux critères : erreur de localisation et temps de localisation), les simulations sont déroulées dans un environnement décentralisé où le résultat obtenu montre que l'algorithme de Bat plus d'efficacité et elle possède une précision meilleurs par rapport aux deux autres algorithmes.

**Algorithme d'optimisation du loup gris (GWO : Gray Wolf Optimization)** est autre méta-heuristique proposé pour le problème de localisation des noeuds [Rajakumar et al., 2017] , GWO est un algorithme imitant quatre types de loups gris pour localiser et chasser des proies, GWO a trois étapes de base : la chasse, la recherche de proies , encerclant sa proie et attaquant sa proie. l'algorithme GWO est implémenté pour la minimisation de l'erreur où la technique RSSI est utilisée pour dans la phase d'estimation de distance, le résultat obtenu montre que GWO converge rapidement et que la localisation est minime, les résultats de simulations sont comparés avec l'algorithme de chauve-souris amélioré et l'algorithme de PSO. alors l'algorithme GWO est plus efficace que les deux autres algorithmes proposés, GWO a prouvé qu'il avait un taux de convergence plus rapide et estime l'emplacement du nœud de sorte que le temps de calcul est minime , en outre , l'algorithme GWO possède des étapes faciles et peu de paramètres à adapter. La question de la localisation présente de nombreux avantages, ce qui n'empêche pas de mentionner des remarques cruciales :

- Le nombre des nœuds est 300 déployé dans un réseau carré de 300 x 300 m, de sorte que la précision des nœuds n'est pas appropriée à la densité du réseau pour juger du problème d'évolution.
- Il est insuffisant de juger de la performance de GWO, car seuls les paramètres des nœuds d'ancre sont évalués dans cet article.
- Les paramètres de GWO ne sont pas évalués.

**Algorithme d'optimisation de la flamme des mites (MFOA :Moth Flame Optimization Algorithm)** est proposé comme une nouvelle technique basé sur la localisation en plusieurs étapes [Mihoubi et al., 2019] ,i.e , la technique est basée sur des nœuds cumulatifs , le nœud localisé à l'itération  $i$  deviendra comme un nœud de référence dans l'itération  $i+1$ , cette stratégie a pour but de renforcer le taux de nœuds de références en même temps diminution de localisation et les erreurs de localisation. Le résultat de la simulation a prouvé que la technique proposée a de bonnes performances avec une supériorité sur la localisation en une étape.

**Algorithme d'optimisation des mouches des fruits (EFOA) amélioré** est proposée pour gérer la localisation dans les réseaux de capteurs sans fils [Mihoubi et al., 2020]. En outre, l'optimisation multicritères est traitée erreur de localisation et le temps de convergence. Le résultat de la simulation est exécuté en deux parties, la première partie, les paramètres de l'algorithme proposé sont évalués sous des scénarios variants, dans la deuxième partie, une étude comparative est menée avec d'autres algorithmes d'optimisations. Le résultat obtenu montre que l'algorithme EFOA donne une erreur de localisation optimale et une convergence rapide.

### 3.8 CONCLUSION

Dans ce chapitre, nous avons décrit les trois phases principales du système de localisation en détail et on a basé beaucoup plus sur la phase de l'algorithme de localisation, dont lequel un état de l'art est étudié, en outre on a fourni une évaluation des différents algorithmes conduits, en se basant sur ses avantages et ses inconvénients. Notant que l'amélioration de localisation peut s'articuler sur n'importe quelle phase du système de localisation, d'où notre thématique est d'améliorer la troisième phase qui est la phase de l'algorithme de localisation, il est mentionné que la localisation possède une catégorisation qui peut aller jusqu'à quatre catégories, par ailleurs, ces différents catégories, on peut les voir comme combinaison les uns aux autres. Dans le chapitre suivant, Algorithme d'optimisation de la flamme des mites sera décrit, discuté et analysé en détail avec les paramètres de configurations de cet algorithme.

# ALGORITHME D'OPTIMISATION DE LA FLAMME DES MITES

# 4

## 4.1 INTRODUCTION

L'optimisation fait référence au processus consistant à trouver la ou les meilleures solutions possibles pour un problème particulier. À mesure que la complexité du problème augmente, au cours des dernières années, le besoin de nouvelles techniques d'optimisations devient plus évident qu'auparavant. Les techniques d'optimisation mathématique étaient les seuls outils d'optimisation des problèmes avant la proposition de techniques d'optimisation heuristique. Les méthodes d'optimisation mathématique sont pour la plupart déterministe et souffrent d'un problème majeur : le piègeage des optima locaux. Certains d'entre eux, comme l'algorithme basé sur le gradient, nécessitent également la dérivation de l'espace de recherche. Cela les rend très inefficaces de résoudre des problèmes réels. Plusieurs algorithmes méta-heuristiques ont été proposées pour pallier les inconvénients susmentionnés des algorithmes déterministes. Au fur et à mesure, d'autres algorithmes ont été proposés tels que, PSO, DE, etc. L'application de ces algorithmes a connu un succès dans différents domaines d'application dans la science et l'industrie. En dépit de tous les avantages de ces optimiseurs, il existe une question fondamentale ici comme s'il y avait un optimiseur pour résoudre tous les problèmes d'optimisation. Selon le théorème No-Free-Lunch (NFL) [Wolpert et Macready, 1997], il n'y a pas un algorithme pour résoudre tous les problèmes d'optimisation. Cela signifie qu'un optimiseur peut bien fonctionner dans un ensemble de problèmes et ne pas résoudre un ensemble de problèmes différents. En d'autres termes, les performances moyennes des optimisations sont égales si l'on considère tous les problèmes d'optimisation. Par conséquent, il existe encore des problèmes qui peuvent être résolus par des nouveaux optimiseurs mieux que les optimiseurs actuels. Face à ce challenge, une recherche considérable a été effectuée pour développer et inspirer d'autres algorithmes pour concurrencer les algorithmes d'optimisation actuels. Dans ce chapitre, un nouveau paradigme d'optimisation inspiré de la nature est proposé, appelé Algorithme d'optimisation de la flamme des mites (MFOA). L'inspiration principale de cet optimiseur est la méthode de navigation des papillons dans la nature appelée orientation transversale. Les papillons volent dans la nuit en maintenant un angle fixe par rapport à la lune, un mécanisme très efficace pour voyager en ligne droite sur de longues distances. Cependant, ces insectes fantaisie sont piégés dans un

chemin en spirale inutile / mortel autour des lumières artificielles. Dans ce chapitre on va voir le modèle mathématique de l'algorithme et le comportement pour effectuer l'optimisation. L'algorithme MFOA est comparé à d'autres algorithmes inspirés de la nature bien connue et aussi sur des problèmes d'ingénierie réels. Les résultats statistiques sur les fonctions de références montrent que cet algorithme est capable de fournir des résultats très prometteurs et compétitifs. De plus, les résultats des problèmes réels démontrent les avantages de cet algorithme pour résoudre des problèmes difficiles avec des espaces de recherches contraintes et inconnues.

## 4.2 ALGORITHME D'OPTIMISATION DE LA FLAMME DES MITES

### 4.2.1 Inspiration

Les mites sont des insectes de fantaisie, qui sont très similaires à la famille des papillons. Fondamentalement, il existe plus de 160 000 espèces différentes de cet insecte dans la nature. Ils ont deux étapes principales dans leur vie : les larves et les adultes. Les larves sont converties en papillons de nuit (mite) par des cocons. Le fait le plus intéressant sur les papillons de nuit (mite) est leurs méthodes de navigation spéciale dans la nuit. Ils ont été développés pour voler dans la nuit en utilisant la lumière de la lune. Ils ont utilisé un mécanisme appelé direction transversale pour la navigation. Dans cette méthode, une mite vole en maintenant un angle fixe par rapport à la lune, un mécanisme très efficace pour parcourir de longues distances en ligne droite [Gaston et al., 2013] [Frank et al., 2006]. **La figure 4.1** montre un modèle conceptuel d'orientation transversale.

La lune étant éloignée du papillon (mite), ce mécanisme garantit un vol en ligne droite. La même méthode de navigation peut être effectuée par l'homme. Supposons que la lune soit dans le côté sud du ciel et qu'un être humain veut aller vers l'est. S'il garde la lune de son côté gauche en marchant, il pourrait se déplacer vers l'est en ligne droite.

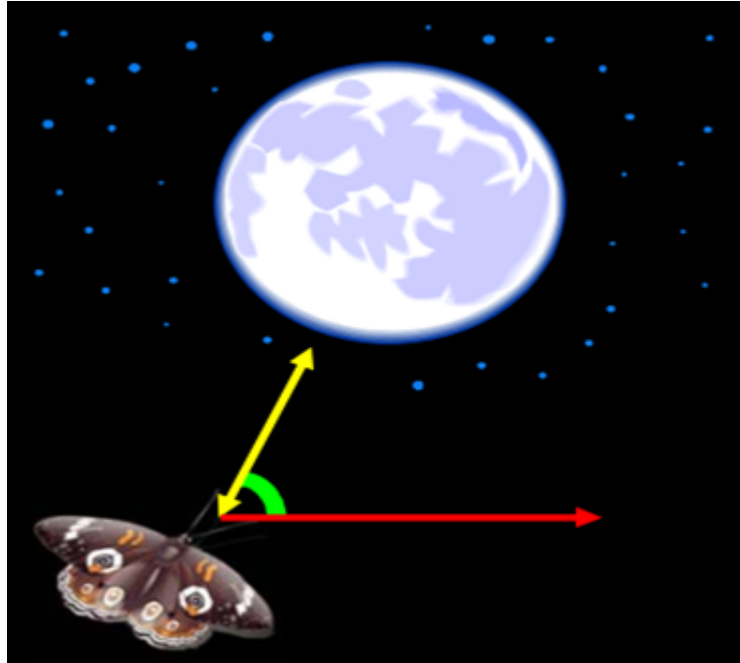


FIGURE 4.1 – Orientation transversale [Mirjalili, 2015c].

En dépit de l'efficacité de l'orientation transversale, nous observons généralement que les papillons volent en spirale autour des lumières. En fait, les papillons sont trompés par des lumières artificielles et montrent de tels comportements. C'est en raison de l'inefficacité de l'orientation transversale, dans laquelle il n'est utile que pour se déplacer dans une ligne droite lorsque la source lumineuse est très éloignée. Lorsque les papillons voient une lumière artificielle, ils essaient de maintenir un angle similaire avec la lumière pour voler en ligne droite. Puisqu'une telle lumière est extrêmement proche par rapport à la lune, cependant, le maintien d'un angle similaire à la source de lumière provoque une trajectoire de vol en spirale inutile ou mortel pour les papillons [Coello, 2000]. Un modèle conceptuel de ce comportement est illustré à la figure 4.2.

À partir de la figure 4.2. On peut observer que le papillon finit par converger vers la lumière. Nous modélisons mathématiquement ce comportement et proposons un optimiseur appelé Algorithme d'optimisation de la flamme des mites (MFOA).

#### 4.2.2 Formulation mathématique

L'algorithme d'optimisation de la flamme des mites (MFOA) est un algorithme métaheuristique basé sur la population, elle est proposée dans [Mirjalili, 2015c]. L'organigramme de l'algorithme est présenté dans la figure 4.3, le MFOA commence la procédure par générer aléatoirement des papillons dans l'espace de solution, ensuite calcule les valeurs de fitness

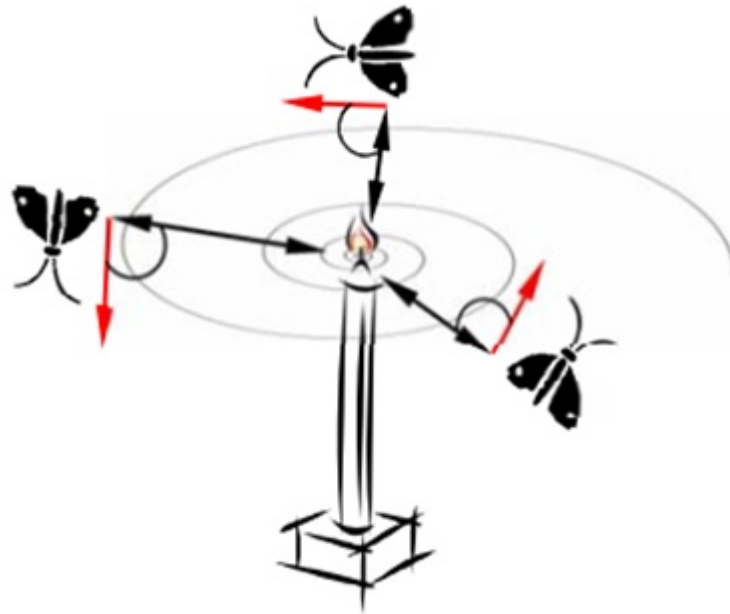


FIGURE 4.2 – *Trajectoire en spirale autour de sources lumineuses .*

(la position) de chaque papillon et à marquer la meilleure position par la flamme.

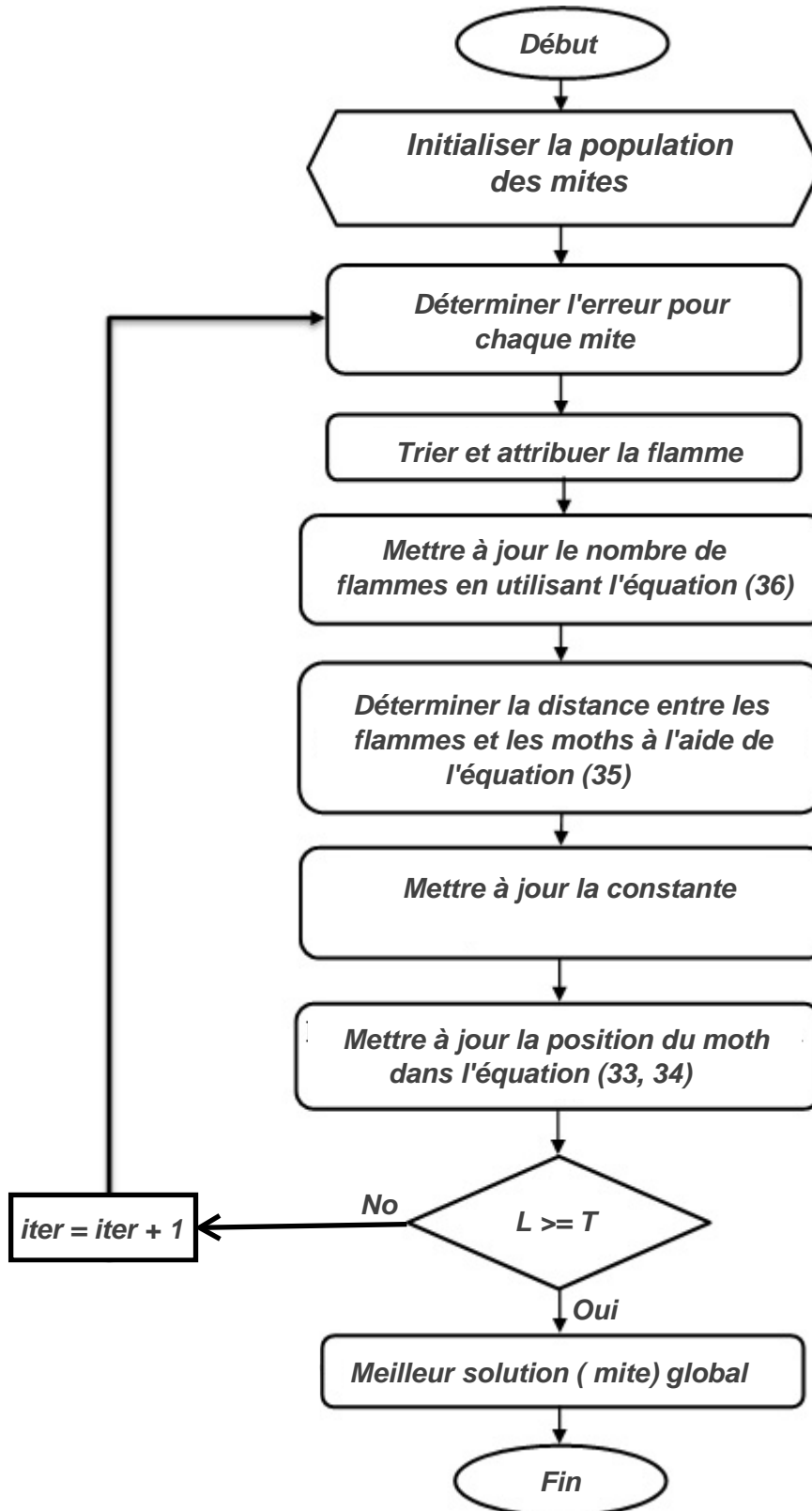


FIGURE 4.3 – Organigramme de l'algorithme MFOA.

En outre, mettre à jour les positions des papillons, sachant que ces positions sont basées sur une fonction de mouvement en spirale pour obtenir de meilleures positions marquées par une flamme, mettre à jour les nou-

velles meilleures positions individuelles et répéter les processus (mettre à jour les positions des papillons et générer de nouvelles positions) jusqu'à ce que les critères d'arrêts sont vérifiées. L'algorithme MFOA est basé sur trois phases fondamentales [Li et al., 2016b]. Ces étapes sont décrites par la suite, par ailleurs, le pseudo-code du MFOA est présenté dans l'**algorithme 1**

### 4.2.3 Génération de la population initiale de papillons

Dans l'algorithme MFOA, nous supposons que les solutions candidates sont des papillons et que les variables du problème sont la position des papillons dans l'espace, par conséquent, les papillons peuvent voler dans un espace d'une seule dimension ou plus, étant donné que l'algorithme MFOA est un algorithme basé sur la population, nous représentons l'ensemble des papillons dans une matrice comme suit :

$$M = \begin{bmatrix} m_{1,1} & m_{1,2} & \dots & \dots & m_{1,d} \\ m_{2,1} & m_{2,2} & \dots & \dots & m_{2,d} \\ \vdots & & \vdots & & \vdots \\ m_{n,1} & m_{n,2} & \dots & \dots & m_{n,d} \end{bmatrix} \quad (23)$$

Où  $n$  est le nombre des papillons et  $d$  est le nombre de variables (dimension). Pour tous les papillons, nous supposons également qu'il existe un tableau pour stocker les valeurs de fitness correspondantes définies comme suit :

$$OM = \begin{bmatrix} OM_1 \\ OM_2 \\ \vdots \\ \vdots \\ OM_1 \end{bmatrix} \quad (24)$$

Notez que la valeur de fitness est la valeur de retour de la fonction de fitness (objectif) pour chaque papillon. Le vecteur de position (représenté par la première ligne dans la matrice  $M$ ) de chaque papillon est transmis à la fonction fitness et la sortie de la fonction fitness est affectée au papillon correspondant comme fonction fitness ( $OM_1$  dans la matrice  $OM$ ). Les flammes sont un autre élément clé de l'algorithme d'optimisation de la flamme des mites. Nous considérons une matrice similaire à la matrice de papillon définie comme suit :

$$F = \begin{bmatrix} F_{1,1} & F_{1,2} & \dots & \dots & F_{1,d} \\ F_{2,1} & F_{2,2} & \dots & \dots & F_{2,d} \\ \vdots & & \vdots & & \vdots \\ F_{n,1} & F_{n,2} & \dots & \dots & F_{n,d} \end{bmatrix} \quad (25)$$

Où  $n$  est le nombre de papillons et  $d$  est le nombre de variables (dimensions). A partir de l'équation 25, on trouve que la dimension des tableaux  $M$  et  $F$  est égale. Pour les flammes, nous supposons également qu'il existe un tableau pour stocker les valeurs de fitness correspondantes définies comme suit :

$$OF = \begin{bmatrix} OF_1 \\ OF_2 \\ \vdots \\ \vdots \\ OF_1 \end{bmatrix} \quad (26)$$

Où  $n$  c'est le nombre des papillons . Il convient de noter ici que les mites et les flammes sont deux solutions, la différence entre eux est la façon dont nous les traitons et les mettons à jour à chaque itération. Les papillons sont de véritables agents de recherche qui se déplacent dans l'espace de recherche, tandis que les flammes sont la meilleure position des mites obtenues. En d'autres termes, les flammes peuvent être considérées comme des drapeaux ou des épingles qui sont lâchés par les papillons lors de la recherche dans l'espace de recherche. Par conséquent, chaque mite recherche autour d'un drapeau (flamme) et le met à jour en cas de trouver une meilleure solution que la solution actuelle. Basant sur ce mécanisme, un papillon de nuit (mite) ne perd jamais sa meilleure solution.

#### 4.2.4 Mise à jour des positions des papillons

L'algorithme MFOA utilise trois fonctions différentes pour faire converger l'optimale globale des problèmes d'optimisation. Ces fonctions sont définies comme suit :

$$MFOA = (I, P, T) \quad (27)$$

Où  $I$  est une fonction qui génère une population aléatoire de mite et les valeurs de fitness correspondantes (les valeurs d'entrée) :

$$I : \Phi \mapsto M, OM \quad (28)$$

La fonction  $P$  fait référence au mouvement des papillons dans l'espace de recherche, elle déplace les papillons dans l'espace de recherche.

$$P : M \mapsto M \quad (29)$$

La fonction  $T$  fait référence à la fin du processus de recherche et donc renvoie la valeur vraie si le critère de terminaison est satisfait et la valeur fausse si le critère de terminaison n'est pas satisfait :

$$T : M \mapsto Vrai, Faux \quad (30)$$

La fonction **I** est utilisée pour générer des solutions initiales et calculer les valeurs des fonctions objectives. Toute distribution aléatoire peut être utilisée dans cette fonction, ce que nous mettons en œuvre est résumé comme suit :

---

```

pour i = 1 : n
pour j = 1 : d
M (i, j) = (ub (i) -lb (i)) * rand () + lb (i);
Fin
Fin
OM = FonctionFitness(M);

```

---

Où :

$$ub = [ub_1, ub_2, \dots, ub_{n-1}, ub_n] \quad (31)$$

$$lb = [lb_1, lb_2, \dots, lb_{n-1}, lb_n] \quad (32)$$

Où  $ub_i$  et  $lb_i$  indiquent les limites inférieure et supérieure de la  $i$ -ème variable, respectivement. Par la suite, la fonction **P** est exécutée de manière itérative jusqu'à ce que la fonction **T** renvoie la valeur vraie. Afin de modéliser mathématiquement ce comportement, nous mettons à jour la position de chaque papillon par rapport à une flamme en utilisant l'équation suivante :

$$M_i = S(M_i, F_j) \quad (33)$$

Où  $M_i$  indique l' $i$ -ème papillon,  $F_j$  indique la  $j$ -ème flamme et **S** est la fonction d'application spirale. Les papillons volent dans l'espace de recherche en utilisant l'orientation transversale. La spirale logarithmique est utilisée comme mécanisme principal de mise à jour des papillons, trois conditions doivent être respectées lors de l'utilisation d'une spirale logarithmique soumise, comme suit :

- Le point initial du spirale devrait commencer à partir du papillon de nuit(mite).
- Le point final de spirale doit être la position de la flamme.
- La fluctuation de la plage de spirale ne doit pas dépasser de l'espace de recherche.

Compte tenu de ces points, nous définissons une spirale logarithmique pour l'algorithme MFOA comme suit :

$$S(M_i, F_j) = D_j * e^{bt} * \cos(2\pi) + F_j \quad (34)$$

Où  $D_j$  indique la distance entre le  $i$ -ème papillon et la  $j$ -ème flamme, **b** est une constante et **t** est un nombre aléatoire choisi dans  $[-1,1]$ . Et la valeur de **D** est calculée comme suit :

$$D_j = |F_j - M_j| \quad (35)$$

Où  $M_i$  représente le  $i$ -ème papillon,  $F_j$  représente la  $j$ -ème flamme . L'équilibre entre l'exploitation et l'exploration est garantie par le mouvement en spirale du papillon de nuit (mite) près de la flamme dans l'espace de recherche. De plus, pour éviter de tomber dans les pièges des optima

locaux, les solutions optimales ont été conservées à chaque répétition et les papillons volent autour de flammes (c'est-à-dire que chaque papillon vole autour de la flamme la plus proche) en utilisant les matrices OF et OM.

#### 4.2.5 Mise à jour du nombre de flammes

La mise à jour des positions des papillons dans  $n$  différents endroits de l'espace de recherche peut réduire les chances d'exploitation des meilleures solutions prometteuses. Pour résoudre ce problème, un mécanisme adaptatif est proposé pour le nombre de flammes, **La figure 4.4** montre que la façon dont le nombre de flammes diminue de manière adaptative au cours des itérations. Par conséquent, la diminution du nombre de flammes aide à résoudre ce problème en fonction de l'équation suivante :

$$Flameno = round(N - l * (N - 1) / T) \quad (36)$$

---

**Algorithme 1** : l'algorithme d'optimisation de la flamme des mites

---

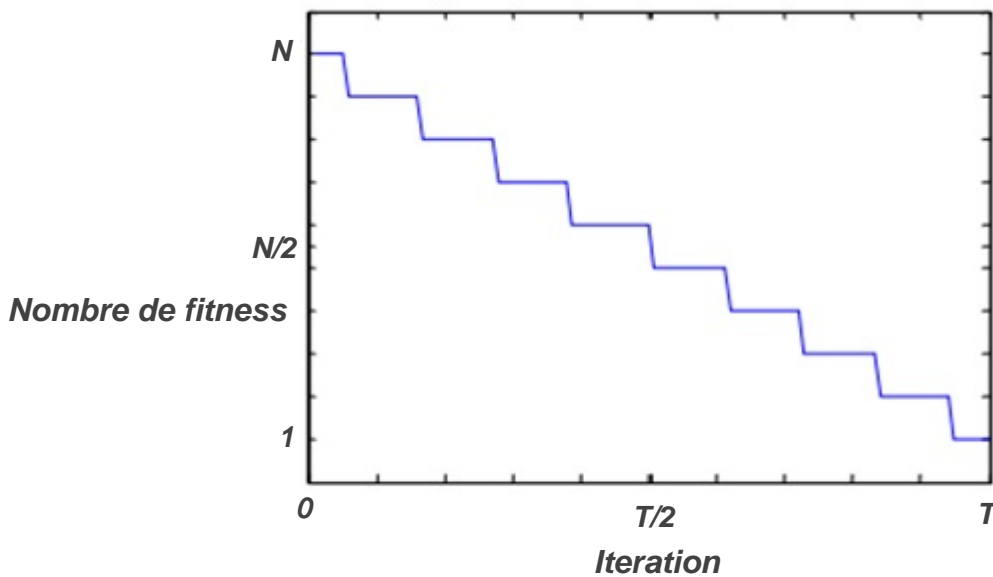
```

initialiser les paramètres de la flamme du papillon
initialiser la position du papillon  $M_i$  au hasard
pour  $i = 1$  à  $n$ 
    calculer la fonction fitness  $f_i$ 
fin pour
Tant que ( l'itération  $\leq$  l'itération maximale )
    - mettre à jour la position de  $M_i$ 
    - calculer le nombre de flammes en utilisant l'équation (36)
    - évaluer la fonction fitness  $f_i$ 
    Si itération == 1 alors
    F = tri (M) et OF = tri (OM)
    Sinon
    F = tri ( $M_{t-1}, M_t$ ) et OF = tri ( $M_{t-1}, M_t$ )
    Fin si
    pour  $i = 1$  à  $n$  do
    pour  $j = 1$  à  $d$  do
    - mettre à jour la valeur de  $r$  et  $t$ 
    - calculer la valeur de  $D$  par rapport à son papillon correspondant
      en utilisant l'équation (35)
    - mettre à jour  $M(i, j)$  par rapport à son papillon correspondant en
      utilisant l'équation ((33) et (34)
    Fin pour
    Fin pour
Fin tant que
Result : afficher la meilleure solution

```

---

Le nombre de flammes diminue de manière adaptative au cours des itérations Où  $l$  est le nombre actuel d'itérations,  $N$  est le nombre maximal de flammes et  $T$  indique le nombre maximal d'itérations. **la figure 4.4** montre qu'il y a  $N$  nombre de flammes dans les étapes initiales des itérations. Cependant, les papillons mettent à jour leurs positions uniquement par rapport à la meilleure flamme dans les étapes finales des itérations. La diminution progressive du nombre de flammes équilibre l'exploration et l'exploitation de l'espace de recherche.

FIGURE 4.4 – *Tri-lateration et Multilateration.*

### 4.3 CONCLUSION

La nature a fourni une vaste source d'inspiration des algorithmes, en particulier les systèmes biologiques. Cela peut à nouveau être utilisé pour développer de nouveaux algorithmes. Dans ce chapitre, l'orientation transversale des papillons a été modélisée pour proposer un nouvel algorithme méta-heuristique basé sur la population. En fait, la convergence en spirale vers les lumières artificielles a été la principale inspiration de l'algorithme MFOA. L'algorithme était équipé de plusieurs opérateurs pour explorer et exploiter efficacement les espaces de recherches. Afin de comparer les performances de l'algorithme MFOA. Noter que l'algorithme MFOA est capable de montrer une exploration élevée et compétitive dans les fonctions multimodales et une exploitation dans les fonctions uni-modales. De plus, les résultats des fonctions de test composite prouvent que l'algorithme MFOA équilibre correctement l'exploration et l'exploitation. La première phase du test a également pris en considération l'observation et la recherche de la convergence de l'algorithme MFOA. Ce chapitre a été consacré à la présentation de l'algorithme d'optimisation de la flamme des mites et à son application adressé au problème d'optimisation. Le problème de localisation des nœuds dans les réseaux de capteurs sans fils a été considéré comme un problème d'optimisation multidimensionnelle. Le prochain chapitre aura pour but de présenter notre contribution pour la localisation des nœuds, la plateforme de simulation et une étude comparative sera effectuée dans plusieurs scénarios de simulations avec les autres approches d'optimisation.

# LOCALISATION DES NOEUDS À BASE DE L'ALGORITHME DE LA FLAMME DES MITES

# 5

## 5.1 INTRODUCTION

La localisation des nœuds dans les réseaux de Capteurs sans Fils est considérée comme un problème d'optimisation complexe, en outre, plusieurs chercheurs ont proposé des techniques opérationnelles pour améliorer la performance du réseau. Le but principal du processus de localisation est de déterminer la position réelle de l'ensemble des nœuds avec un coût minimal et dans un temps optimal. Dans ce chapitre plusieurs paramètres de performance ont été étudiés et évalués tel que le rayon de communication, la densité des nœuds de références, le taux de la population. par ailleurs, l'évaluation de performance est basée sur deux paramètres de métriques : le temps nécessaire de localisation et l'erreur de localisation moyenne. Pour optimiser la performance de la localisation dans les réseaux de capteurs sans fils, une étude comparative a été conduite entre l'approche proposée et les algorithmes métaheuristiques les plus populaires et les plus récents dans ce domaine. L'étude comparative élaborée s'articule principalement sur les deux critères d'évaluation à optimiser : le temps de localisation moyenne et l'erreur de localisation moyenne. L'ensemble des simulations établies dans la plateforme associée et basée sur un langage de programmation MATLAB.

## 5.2 FORMULATION DU PROBLÈME DE LOCALISATION

L'objectif de la localisation dans les réseaux de capteurs sans fils est d'estimer les coordonnées de  $N$  noeuds inconnus sur la base de  $M$  noeuds de références. En supposant que tous les noeuds de capteurs sont déployés dans un champ de capteur bidimensionnel. Le processus de localisation basée sur l'algorithme d'optimisation de la flamme des mites est illustré dans la figure 5.1 .

Par ailleurs, la localisation des noeuds dans les RCSFs consiste à calculer les coordonnées des noeuds inconnus (noeuds cibles) en utilisant les informations inter-capteurs gérées par les noeuds de références.

Les étapes fondamentales du processus de localisation sont résumées dans les points suivants :

1. L'ensemble des noeuds ( $N$  noeuds de références et  $M$  noeuds inconnus) sont déployés de manière aléatoire dans l'environnement de captage, associés à  $N$  et  $M$  le même rayon de transmission puis définir la plage de communication ( $R$ ) pour chaque noeud de capteur.
2. Les noeuds de références diffusent fréquemment leurs positions.
3. chaque noeud inconnu qui est entouré (dans sa plage de communication) par trois noeuds de référence ou plus , donc il est considéré comme un noeud localisable, cela est grâce à la technique de RSSI et avec une mise en œuvre simple et un faible coût matériel dans le scénario de déploiement réel.

en outre,nous supposons que la mesure de la distance entre les noeuds voisins est basée sur la technique de RSSI qui peut ensuite être transféré à des distances équivalentes pour le positionnement de l'intersection [Patwari et al., 2005] [Vecchio et al., 2012] .

la distance réelle  $d_{ij}$  entre un noeud inconnu  $(x_i, y_i)$  et le  $j$ -ième noeud de référence  $(x_j, y_j)$  est définie par l'équation suivante :

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (37)$$

La distance mesurée  $d_{ij}$  entre un noeud inconnu  $(x_i, y_i)$  et le  $j$ ème noeud de référence  $(x_j, y_j)$  est modélisée à l'aide de l'équation suivante :

$$\widehat{d}_{ij} = [d_{ij} + n_{ij}] \quad (38)$$

Où  $n_{ij}$  représente l'erreur comprise entre un noeud inconnu  $(x_i, y_i)$  et son noeud de référence voisin  $(x_j, y_j)$ .

4. Établir la fonction objective  $f(x_i, y_i)$  , la fonction objective représente l'erreur quadratique moyenne entre un noeud inconnu et les noeuds de références correspondants, elle est définie par l'équation suivante :

$$f(x, y) = \frac{1}{N} \sum_{i=1}^N (\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} - \widehat{d}_{ij})^2 \quad (39)$$

$$f(x, y) = \frac{1}{N} \sum_{i=1}^N (d_{ij}^2 - \widehat{d}_{ij})^2 \quad (39)$$

Où  $N$  ( $N > 3$ ) représente le nombre de noeuds de références dans la plage de communication, le noeud cible peut estimer ses coordonnées en exécutant l'algorithme d'optimisation de la flamme des mites. En outre, une fois

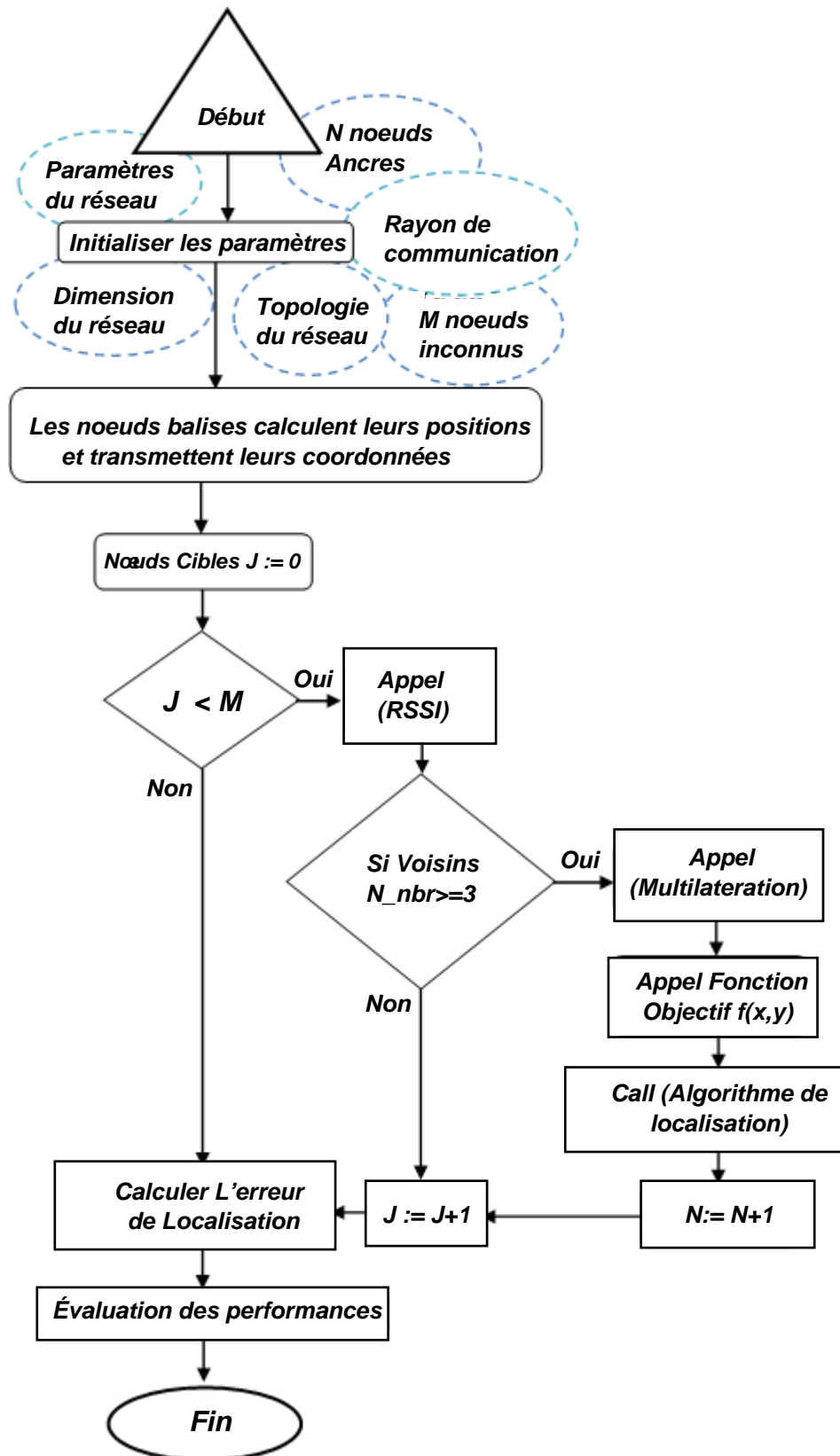


FIGURE 5.1 – Organigramme de la localisation des noeuds.

la fonction objective est minimisée, alors  $(x_i, y_i)$  représente la position du

$i^{me}$  nœud inconnu.

5. Les nœuds inconnus localisés (ciblés) agissent (participent) comme des nœuds de référence lors de la prochaine itération (opération), de sorte que le nombre de nœuds de référence augmente avec la progression de l'itération et le nombre de nœuds inconnus diminue.

6. Les étapes 2 à 5 sont répétées jusqu'à ce que la condition d'arrêt soit atteinte (aucun nœud inconnu ne peut être localisé ou nombre d'itérations effectuées ou erreur de tolérance respectée).

7. Calculer l'erreur de localisation moyenne (MLE), l'erreur de localisation moyenne est établie comme la distance euclidienne moyenne entre les emplacements réels et les emplacements estimés des nœuds de capteurs. ainsi l'erreur de localisation moyenne peut être calculée à l'aide de l'équation suivante :

$$E_L = \frac{1}{M_L} \sum_{i=1}^L (\sqrt{(x_i - X_i)^2 + (y_i - Y_i)^2}) \quad (40)$$

Où  $M_L$  est le nombre de nœuds localisés,  $(x_i, y_i)$  est l'emplacement du nœud calculé et  $(X_i, Y_i)$  est l'emplacement du nœud réel. Plus l'erreur de localisation moyenne est petite, plus les performances de localisation sont meilleures.

**Où "L = M-N", M est le nombre de nœuds localisés.**

À mesure que le nombre d'itérations progresse alors le nombre de nœuds localisés augmente. Cela augmente le nombre des noeuds d'ancres comme référence disponible pour les nœuds déjà localisés, un nœud localisé via trois ancres dans l'itération actuelle peut avoir plus de référence dans la prochaine itération.

### 5.3 ÉVALUATION DES PERFORMANCES

Dans cette section, l'évaluation des performances de l'algorithme d'optimisation de la flamme des mites a été effectuée sur la base des paramètres suivants :

- La densité des nœuds d'ancrage.
- La portée de communication.
- La taille de la population.

En outre, une étude comparative a été menée entre l'approche proposée et les algorithmes méta-heuristiques les plus populaires et les plus récents comme : PSO, DE, BAT, FPA et GWO, en notant que des tests comparatifs sont réalisés en utilisant le même déploiement du réseau.

#### 5.3.1 La métrique d'évaluation

Afin d'évaluer les performances de l'approche proposée, deux métriques majeures [Abdi et Haghghat, 2014] sont proposées, de plus, ces métriques sont ensuite utilisées pour établir une étude comparative avec d'autres algorithmes méta-heuristiques :

- **Erreur de Localisation Moyenne (ELM)** : l'erreur de la localisation est définie comme la distance euclidienne moyenne entre les emplacements réels et les emplacements estimés des nœuds de capteur. (les coordonnées du nœud (L'emplacement estimé)  $(x_i, y_i)$  et les coordonnées du nœud réel  $(X_i, Y_i)$  ). ELM est calculée à l'aide de l'équation suivante (41).

$$E_L = \frac{1}{M_L} \sum_{i=1}^L (\sqrt{(x_i - X_i)^2 + (y_i - Y_i)^2}) \quad (41)$$

• **Le Temps de localisation Moyen** : est défini comme le temps de simulation nécessaire pour localiser l'ensemble des nœuds qui ont réussi à obtenir leur emplacement sur le nombre total de nœuds capteurs, TLM est calculé à l'aide de l'équation suivante (42).

$$\text{Temps de localisation Moyenne} = \frac{\sum_{i=1}^n \text{Temps d'execution}}{n(\text{nombre de noeud de capteur})} \quad (42)$$

Notez que, plus le temps de localisation est minimal, plus les performances de localisation sont meilleures et cela permet de conserver beaucoup plus l'énergie des nœuds de capteurs et donc la durée de vie du réseau.

### 5.3.2 Configuration de la simulation

Les simulations sont effectuées en utilisant l'environnement MATLAB sur un ordinateur portable de 8 Go de mémoire et un processeur i5, 2.40 GHz. En outre, dans nos simulations, nous avons supposé les hypothèses suivantes :

- Les nœuds sont considérés comme homogènes en ce qui concerne la puissance de l'émetteur-récepteur et la sensibilité du récepteur. Cela permet de contrôler facilement la connectivité entre les nœuds du réseau.
- Pour simplifier, nous considérons la plage de transmission des nœuds comme un cercle parfait. Cela garantit que les paquets transmis par un nœud voisin sont toujours reçus avec succès.
- Tous les nœuds de capteur sont synchronisés.
- Le rayon de communication utilisé pour les nœuds capteurs (ancres ou inconnus) est égal (dans le même scénario) pendant tous les scénarios de simulation.

La distance estimée entre les nœuds inconnus et les nœuds de référence a été calculée par l'équation suivante :

$$P_r(d) = P_0(d_0) - 10 * \eta_p * \log_{10}\left(\frac{d}{d_0}\right) + X_\sigma \quad (43)$$

Où  $d$  représente la distance entre l'émetteur et le récepteur.

$d_0$  représente la distance de référence et est généralement égale à un mètre.

$P_r(d)$  représente la puissance du signal reçu.

$P_0(d_0)$  représente la puissance d'émission.

$X_\sigma$  représente une variable aléatoire distribuée gaussienne moyenne nulle dont la valeur moyenne est nulle et reflète le changement de puissance du signal reçu.

$\eta_p$  représente l'exposant de perte de trajet qui mesure la vitesse à laquelle le RSSI diminue avec la distance.

l'ensemble des paramètres du réseau sont décrits dans la **table 5.1**, **Par ailleurs, nous récapitulons les paramètres de simulation comme suit :**

- les nœuds sont déployés de manière aléatoire dans un champ de capteur d'une superficie variée durant la simulation entre [50 x 50 m<sup>2</sup> — 100 x 100 m<sup>2</sup>].
- la communication symétrique et la portée de communication sont variées dans l'intervalle suivant : [5, ..., 60] mètres.

- la densité des noeuds d'ancrages est variée dans l'intervalle [10% ..., 80%]. Nous définissons la densité d'ancrage comme le rapport entre les noeuds d'ancrages et le nombre total de noeuds dans le réseau.
- la puissance d'émission est de -5 dBm.

TABLE 5.1 – Configuration des paramètres du RCSF

Paramètre	Valeur
Noeuds de capteurs	50 , 100
Noeuds de références	$\sum_{i=1}^{20} 4 * i$
Zone de déploiement ( $m^2$ )	(50 * 50) , ( 100 * 100)
Portée de transmission (mètres)	$\sum_{i=1}^{12} 5 * i$
Nombre maximum d'itérations	$\sum_{i=1}^{10} 10 * i$

L'ensemble des noeuds de capteurs sont déployés de manière aléatoire dans une zone rectangulaire bidimensionnelle de (100 x 100  $m^2$ ) à l'aide d'un générateur pseudo-aléatoire à distribution uniforme continue. Le nombre total des noeuds de capteurs, à savoir la densité des noeuds, devrait être de 100, 200 et 300. Les noeuds d'ancrages représentent 10% à 80% de l'ensemble des noeuds de capteurs. Les noeuds d'ancrages sont déployés de manière aléatoire. On suppose qu'il n'y a pas d'erreur de localisation pour les noeuds d'ancrages. La plage de communication est définie sur 5 mètres à 60 mètres. Pour éliminer les effets du caractère aléatoire de la génération de la topologie et de l'algorithme bio inspiré, chaque point de données est moyennement exécuté sur 20 réseaux de tests différents et chaque résultat pour une sorte de topologie du réseau est aussi moyennement exécuté 30 fois de manière répétée. Des expériences approfondies ont été menées pour concevoir les paramètres de l'algorithme d'optimisation de la flamme des mites d'où les résultats expérimentaux démontrent qu'une précision de localisation plus élevée avec moins d'itérations peut être obtenue lorsque le réglage des paramètres de l'algorithme est le suivant :

1 -  $a_{min}$  et  $a_{max}$  sont définis sur 0,9 et 1,0 respectivement.

2 -  $Pa_{min}$  et  $Pa_{max}$  sont définis sur 0,05 et 0,25 respectivement.

Pour empêcher la longueur de la recherche sur le champ du réseau, la limite du champ est définie pour être la plage de chaque imbrication. Ainsi, dans une zone de bidimensionnelle comme exemple 5 x 50  $m^2$  et 100 x 100  $m^2$ , la valeur minimale ( $X_{min}$ ) et la valeur maximale ( $X_{max}$ ) sont définies sur 0 et 50 et sur 0 et 100 ,respectivement. **La figure 5.2** montre le déploiement des noeuds dans une zone de captage bidimensionnelle.

les noeuds verts représentent les noeuds Ancres et les noeuds bleus représentent les noeuds inconnus. Le taux des noeuds Ancres est de 20% de l'ensemble des 50 noeuds. Le diagramme de relation de voisinage associé au déploiement est présenté dans **la figure 5.3**, où le rayon de communi-

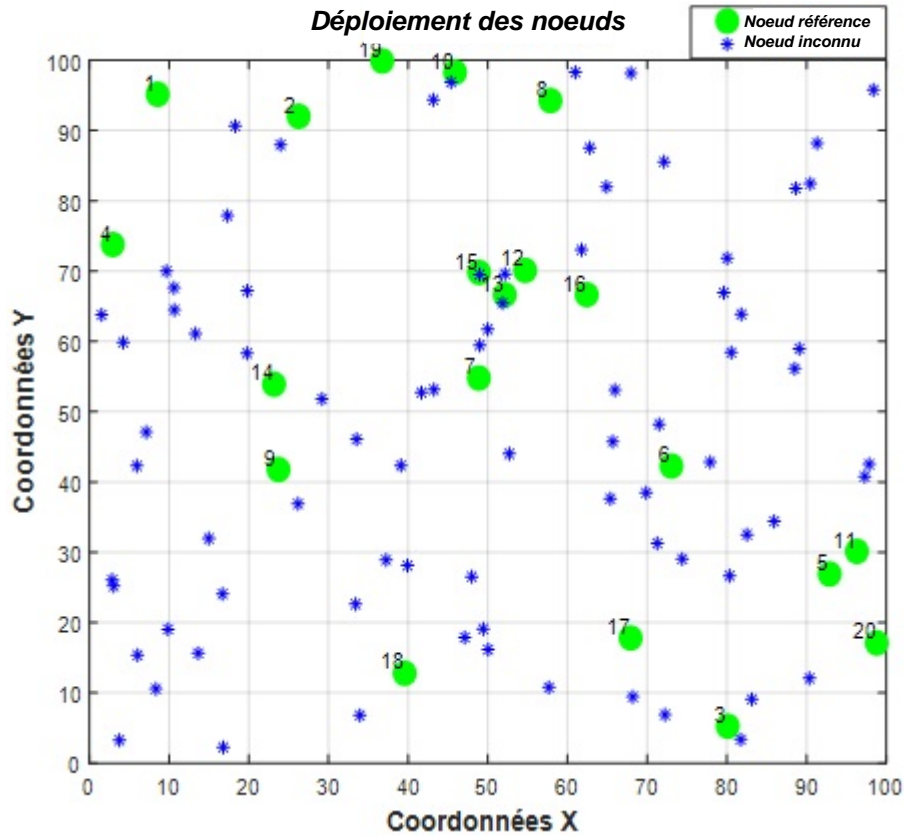


FIGURE 5.2 – Déploiement des noeuds dans le réseau.

cation est de 30 mètres. Le diagramme de relation de voisinage explique si les noeuds inconnus sont dans l'audience des noeuds d'ancrages ou non.

La position exacte des noeuds cibles estimée par l'algorithme d'optimisation de la flamme des mites est présentée dans **la figure 5.4**, la forme du triangle représente les noeuds résolus. A partir de cette figure la, nous constatons que les noeuds localisés sont peu parce que la gamme de communication est insuffisante pour couvrir tous les noeuds du réseau.

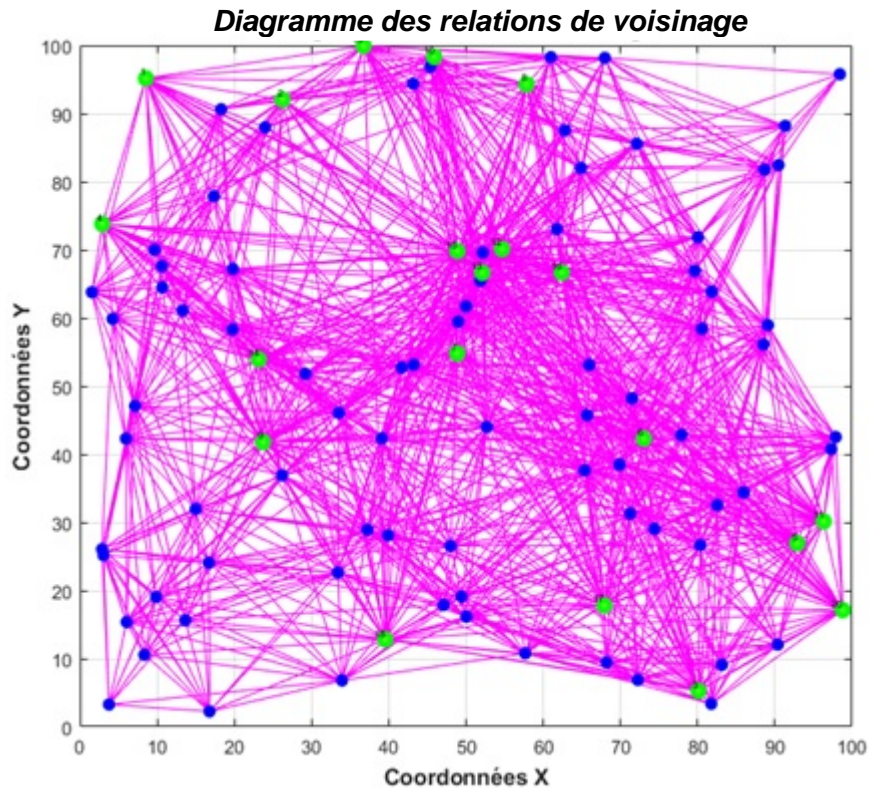


FIGURE 5.3 – Connectivité de communication entre les noeuds.

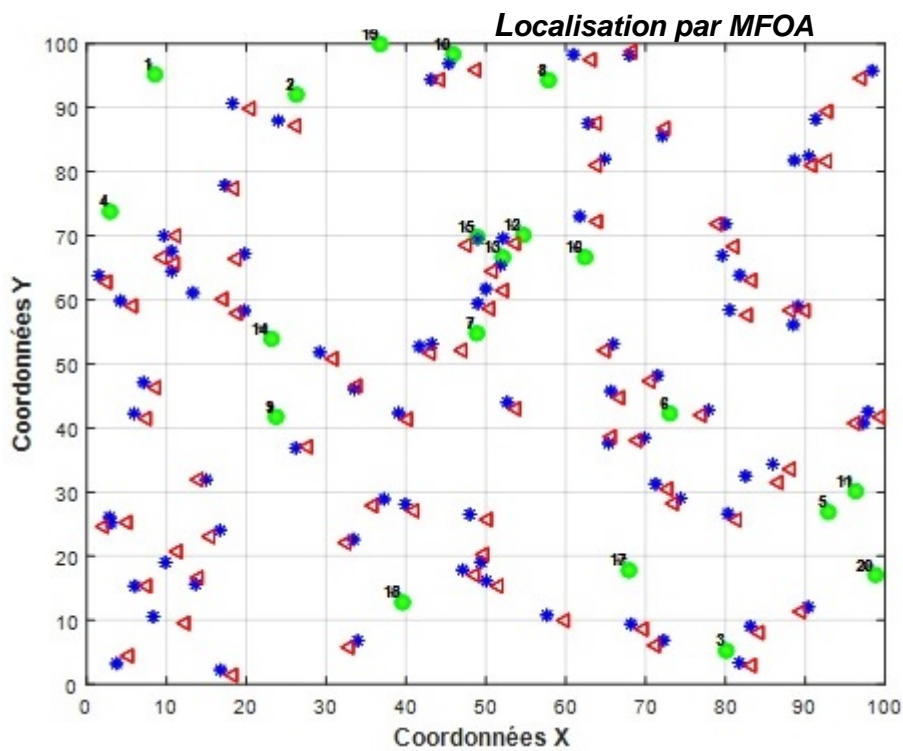


FIGURE 5.4 – Localisation des noeuds par l'algorithme MFOA .

### 5.3.3 Résultats expérimentaux et analyse des performances

Cette section est principalement consacrée à l'étude des paramètres de la simulation. Nous analysons les influences plausibles sur le taux de localisation et les paramètres comprennent la taille de la population, le rayon de la communication et la densité des noeuds de références .

### 5.3.4 L'effet de la densité des nœuds d'ancrages

La densité des noeuds de références est un paramètre important affectant les performances de localisation et le coût pour les réseaux de capteurs sans fils. Dans cette partie de ce chapitre et pour objectif de déterminer l'influence de ce paramètre, plusieurs expériences ont été réalisées et les effets de la densité des noeuds de référence sur les performances de localisation sont évalués.

La densité des nœuds de référence est variée dans l'intervalle [10%... 80%] de l'ensemble des nœuds de capteurs. La portée de communication de chaque nœud de capteur est fixée à une valeur de 30 mètres.

La **figure 5.5** représente l'erreur de localisation obtenu durant le changement du taux de densité des nœuds ancrages basant sur l'algorithme d'optimisation de la flamme des mites , à partir de **la figure 5.5** on trouve que l'erreur de localisation moyenne et l'intervalle de confiance de l'erreur de localisation sont évalués en faisant varier le rapport d'ancrage dans le réseau pour les différentes densités des nœuds. L'intervalle de confiance représente une estimation de la plage de l'erreur de localisation moyenne. Il est clairement observé que lorsque le taux des noeuds d'ancrages augmente de 10% à 40%, la précision de localisation s'améliore considérablement. Parce qu'avec le nombre croissant des nœuds d'ancrages, le nombre des nœuds inconnus ( qui peuvent réaliser une localisation basée sur des nœuds d'ancrages d'origines (les nœuds sont localisés par des récepteurs GPS ou un déploiement manuel)) augmente. Cependant, lorsque le rapport d'ancrage continue de croître, les effets sur l'erreur de localisation moyenne deviennent très minimes et insignifiants. Cela signifie qu'il n'est pas nécessaire parfois d'augmenter le nombre des nœuds d'ancrages et cela nécessite un matériel supplémentaire spécifique qui peut être coûteux. De plus, à mesure que la densité des nœuds augmente, l'erreur de localisation moyenne diminue en conséquence.

Les résultats de la simulation présentés par **la figure 5.5** montrent que l'intervalle de confiance est évidemment élargi lorsque le rapport d'ancrage est de 10% et la densité des nœuds est de 100. Cela est attribué au fait qu'il y a moins noeuds de références dans la plage de communication et la propagation d'erreur se produit entre les nœuds de capteurs, ce qui a un impact négatif sur l'erreur de localisation moyenne.

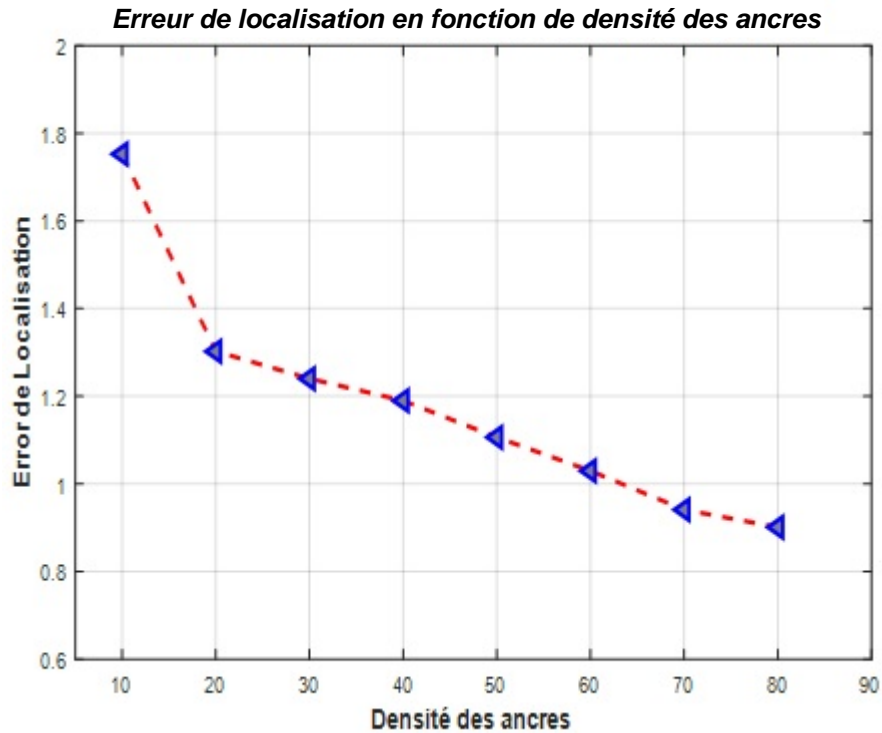


FIGURE 5.5 – L'effet du rapport d'ancrage sur l'erreur de localisation moyenne.

Dans l'autre côté, l'évolution du temps de localisation par apport au nombre d'itérations est présenté dans la **figure 5.6**, nous observons que , quand la densité des nœuds de références est de 10 % , le temps de localisation est considérable, en effet, plus le nombre des nœuds de références augmente plus le temps de localisation diminue, cela est obtenu par la technique incrémentale ou itérative , cela veut dire que ,tout nœud inconnu localisé dans l'itération actuelle, il sera ajouté à la liste des nœuds ancrés ( considéré comme nœud de références) et donc participe à la localisation dans la prochaine itération .

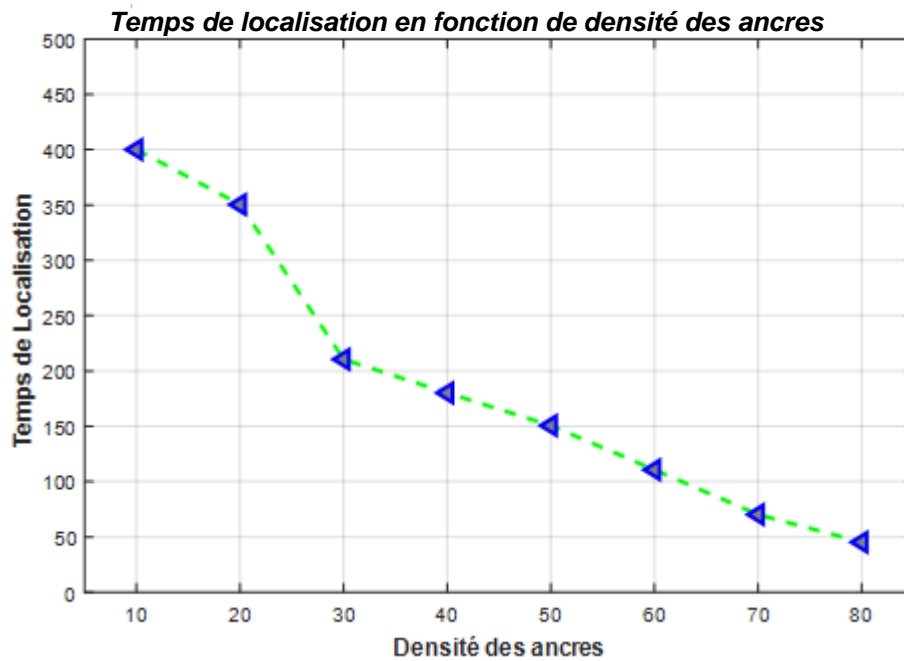


FIGURE 5.6 – L'effet du rapport d'ancrage sur le temps de localisation moyenne.

### 5.3.5 L'effet de la portée de communication

La plage (portée) de communication est un autre paramètre très important qui détermine les performances de la localisation et beaucoup plus la consommation d'énergie des nœuds de capteurs. L'impact de la portée de communication sur l'algorithme de localisation appliquée (l'algorithme d'optimisation de la flamme des mites) est illustré dans la figure 5.7. l'évaluation est faite en matière d'erreur de localisation moyenne sous différentes valeurs de densités de nœuds .

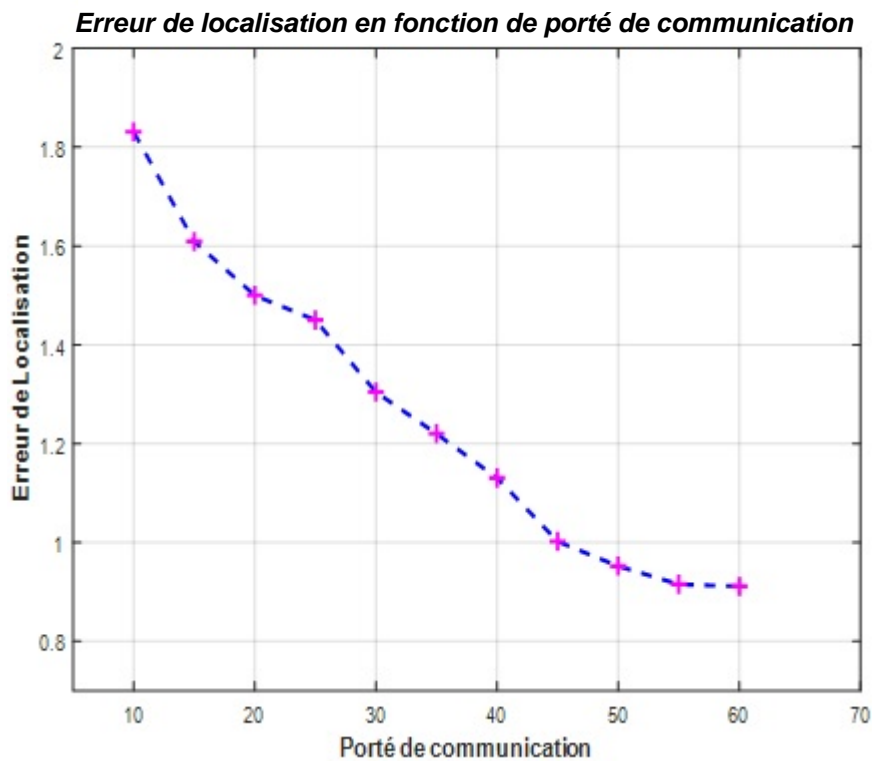


FIGURE 5.7 – L'effet de la portée de communication sur l'erreur de localisation moyenne.

Le nombre des nœuds de références est défini à un taux de 20% de l'ensemble des nœuds de capteurs. La variation de la portée de communication est exécutée dans l'intervalle qui s'écoule de 10 m à 60 m. A partir de **la figure 5.7**, nous constatons que lorsque la plage de communication est inférieure à 20 mètres, la relation est médiocre entre les nœuds et donc l'erreur de localisation moyenne est légèrement plus importante. Cela est dû au fait que le réseau n'est pas fortement connecté pour une grande partie des nœuds, la connectivité entre les nœuds est présentée dans les figures **la figure 5.8**, **la figure 5.9**, **la figure 5.10**, **la figure 5.11**.

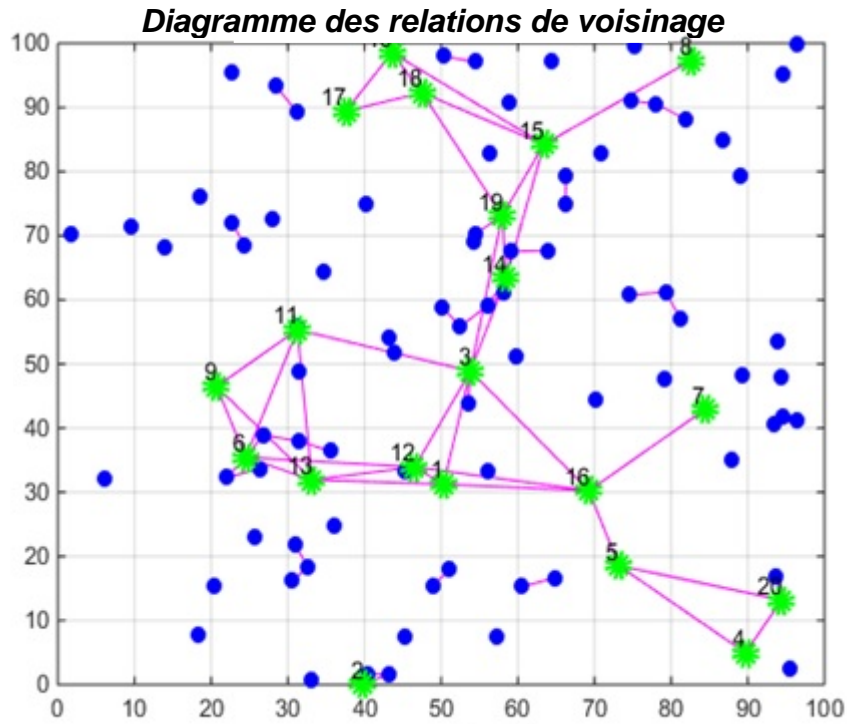


FIGURE 5.8 – Connectivité de communication entre les nœuds de capteurs (10 mètres)

On constate que lorsque la portée de communication augmente, l'erreur de localisation moyenne diminue considérablement. Cela est dû au fait qu'il existe plus d'informations d'ancrages disponibles pour calculer l'emplacement des nœuds inconnus.

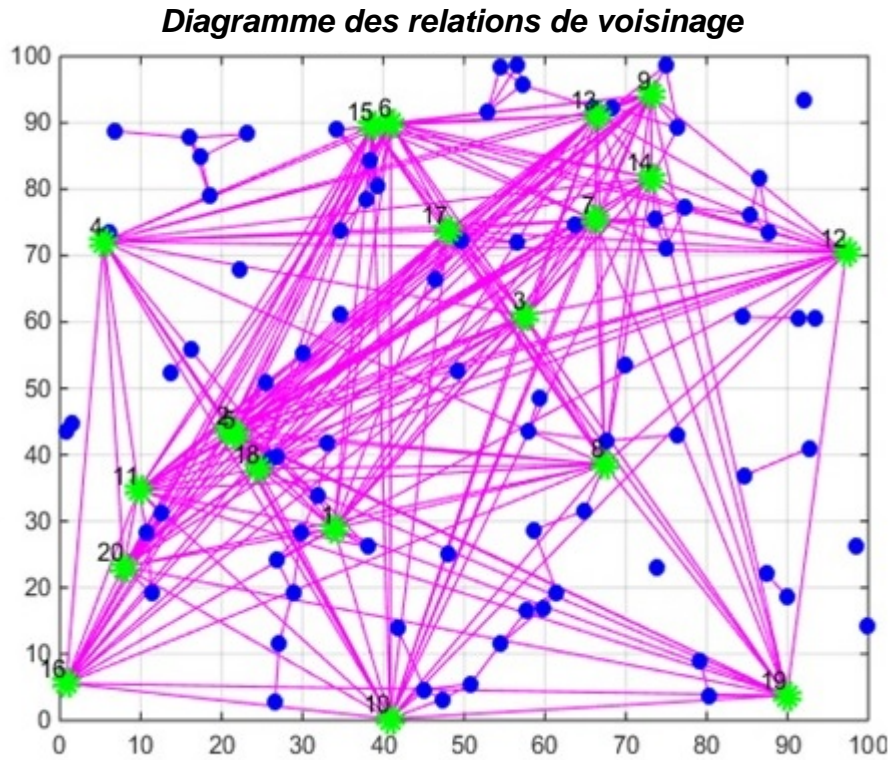


FIGURE 5.9 – Connectivité de communication entre les nœuds de capteurs (20).

Ainsi, lorsque la plage de communication augmente jusqu'à une certaine valeur suffisante, alors l'erreur de localisation moyenne diminue légèrement cela est dû parce qu'il y a plus d'informations d'ancrages disponibles pour calculer l'emplacement des nœuds inconnus.

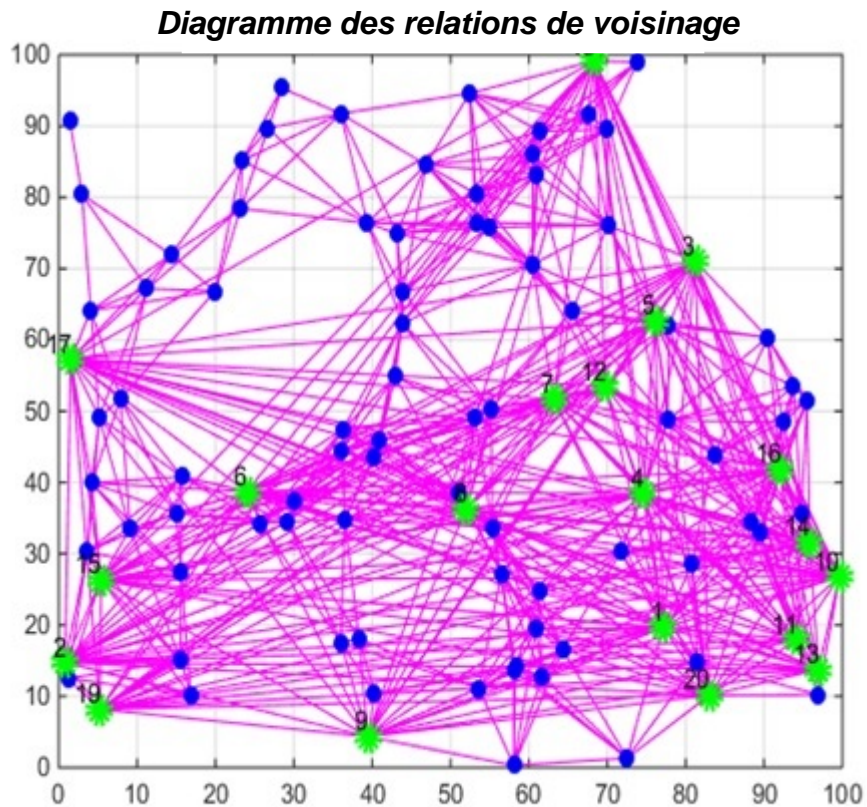


FIGURE 5.10 – Connectivité de communication entre les nœuds de capteurs (30).

La précision de localisation dépend également de la densité des nœuds. On peut observer clairement que l'erreur de localisation moyenne diminue progressivement par rapport à l'augmentation de la densité des nœuds. Cela est dû au fait que lorsque la densité de nœuds augmente, la connectivité entre les nœuds devient élevée et le nombre de nœuds d'ancrages disponibles dans la plage de communication augmente, cet événement est interprété par le fait que les nœuds sont bien dispersés dans le réseau, et qu'ils peuvent rejoindre plusieurs nœuds pour participer à l'opération de la localisation, par ailleurs, elles dépendent également du déploiement initial des nœuds de capteurs. D'autre part, lorsque le rayon de communication est plus élevé et couvre une grande région, alors la relation sera forte (voir la figure 5.11).

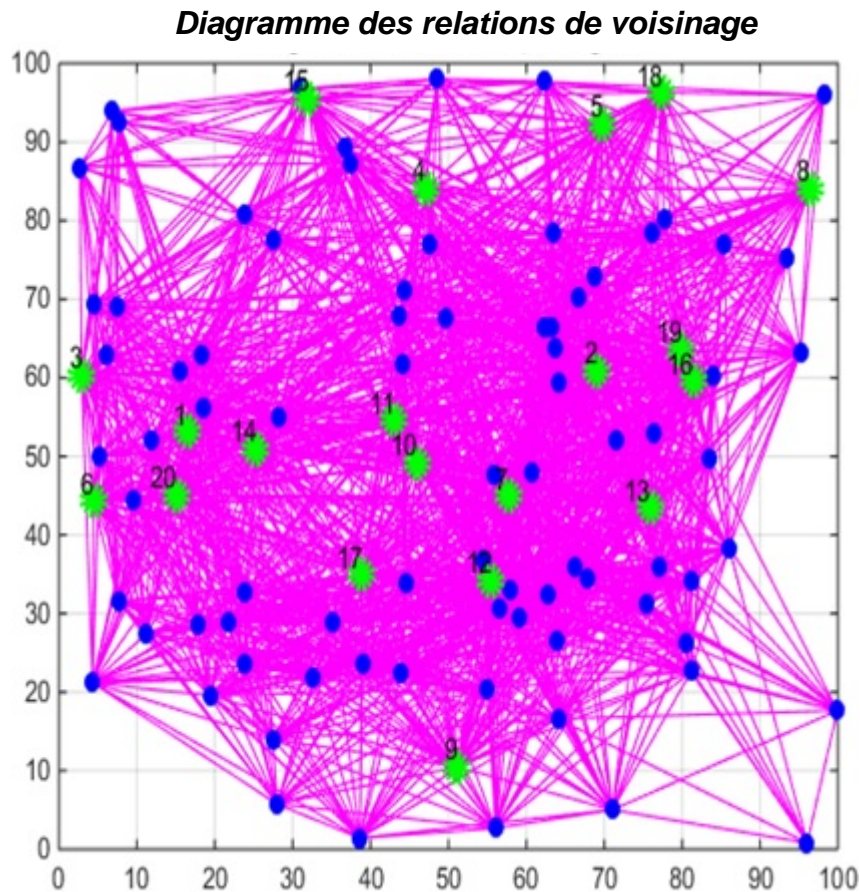


FIGURE 5.11 – Connectivité de communication entre les nœuds de capteurs (40).

Dans l'autre côté, l'évolution de la plage de communication vers le temps de la localisation est présenté dans la **figure 5.12**, nous observons que l'algorithme d'optimisation de la flamme des mites a une vitesse de convergence régulière, cela est dû au fait que lorsque la plage de communication est courte, il y a moins de nœuds à localiser autour des nœuds ancrés et la zone de couverture est assez insuffisante, mais l'erreur de localisation est grande. En effet, lorsque la portée de communication augmente, le temps de localisation devient considérable parce que le signal envoyé prend plus de temps pour joindre sa destination et couvre le maximum des nœuds inconnus. sachant que le bruit de l'environnement influence négativement sur la localisation des nœuds et donc augmente l'erreur de localisation. Nous concluons que le temps de localisation est meilleur quand la plage de communication est suffisante pour avoir une erreur de localisation minimale, et cela permet de conserver la consommation d'énergie du réseau.

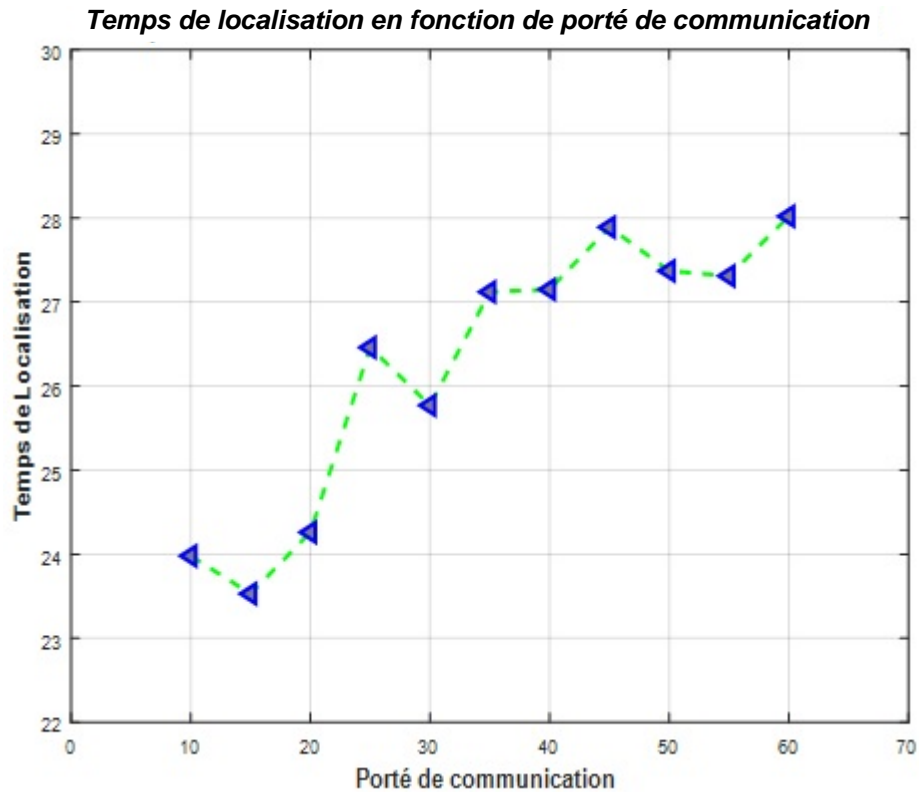


FIGURE 5.12 – L'effet de la portée de communication sur le temps de localisation moyenne.

### 5.3.6 L'effet de la taille de population

La taille de la population est considérée comme un facteur très essentiel pour la convergence de l'algorithme, principalement dans la phase d'exploration de recherche des voisins. De plus, la taille de la population affecte la solidité et la durabilité du réseau et donc influence sur la consommation d'énergie des nœuds et globalement sur la durée de vie du réseau. La taille de population choisie pour un tel domaine d'application n'est pas la même [GomesPereira de Lacerda et al., 2018] [Bansal, 2020], et cela revient à beaucoup critères comme : la complexité du problème, la densité des variables etc. Les résultats de l'évolution de l'erreur de localisation vers la taille de la population sont présentés dans la figure 5.13.

A partir de la figure 5.13, on trouve que l'algorithme converge lorsque le taux de population est important. Cette caractéristique est avantageuse pour l'algorithme dans la phase d'exploration. d'où, le processus de génération de la solution candidate autour de la meilleure solution augmente la capacité d'exploitation.

Dans l'autre côté, les résultats de l'évolution du temps de localisation vers la taille de la population sont présentés dans la figure 5.14. A partir de la figure 5.14, nous trouvons que avec une taille de population petite, le temps de localisation est minimale et avec l'augmentation de la population donc le temps de localisation devient considérable, cela est dû que lors de la phase d'exploitation de l'algorithme, l'algorithme exploite tous les

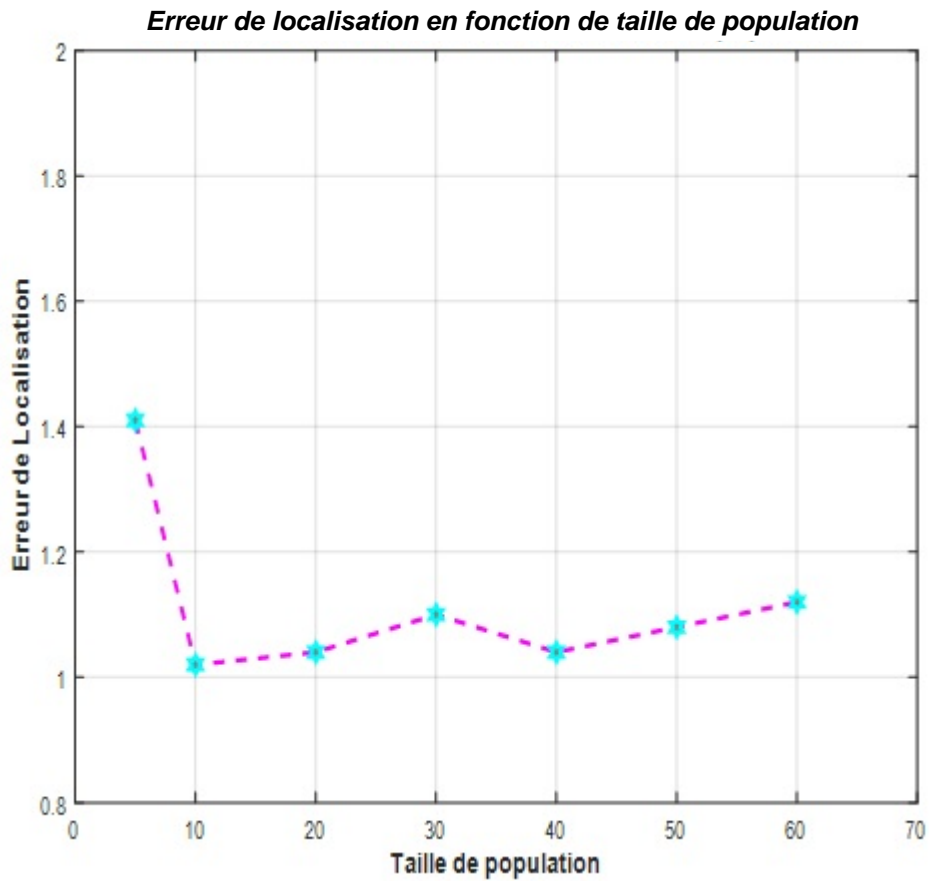


FIGURE 5.13 – Erreur de localisation par rapport à la taille de la population.

voisins pour obtenir les meilleures coordonnées, donc plus le nombre de population est grande, plus le temps de parcourir les voisins est long.

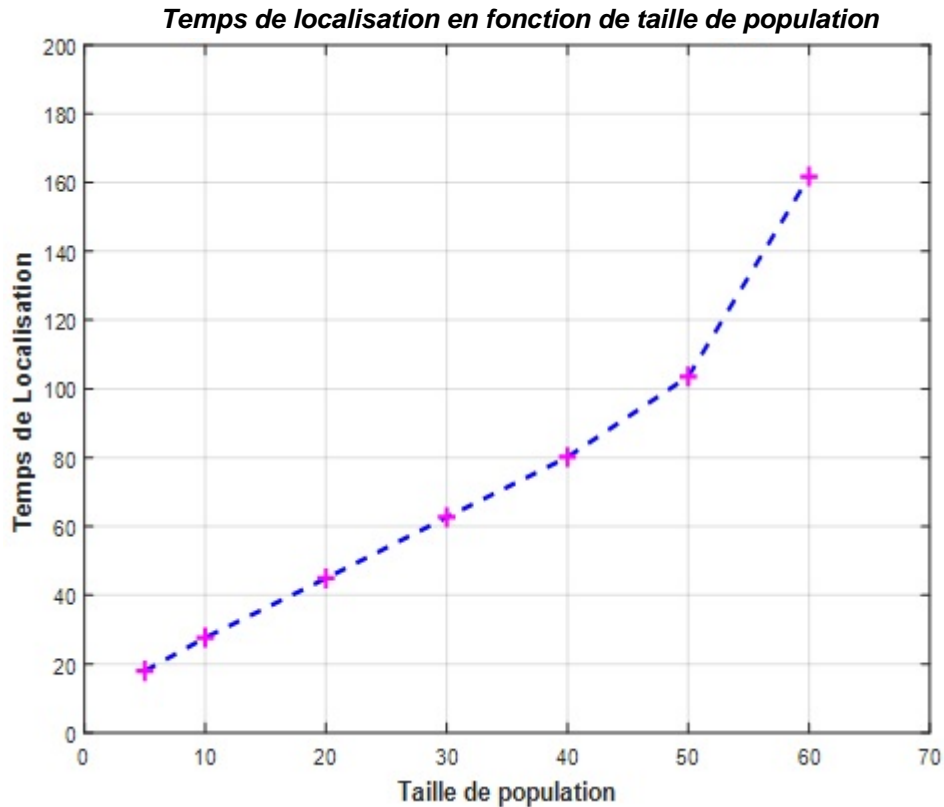


FIGURE 5.14 – Temps de localisation par rapport à la taille de la population.

### 5.3.7 Etude comparative

Cette partie de ce chapitre est consacrée pour établir une étude comparative appliquée au problème de localisation des nœuds dans les réseaux de capteurs sans fils, cette étude est basée sur l'approche proposée avec les autres méta-heuristiques les plus populaires et aussi les plus récentes dans ce domaine. En outre, pour éviter une recherche insignifiante sur le champ du réseau, les bornes l'algorithme d'optimisation de la flamme des mites sont définis  $X_{min}$  égale à 0 et  $X_{max}$  égale à 50. Le déploiement du réseau est présenté dans la figure 5.15. le déploiement des nœuds est effectué aléatoirement dans une zone de 50 m x 50 m (2-D) . Le nombre total des nœuds est de 60 nœuds, dont 56 nœuds représentent les nœuds inconnus, pour but d'éviter le problème colinéaire , alors quatre nœuds de références (couleur verte) sont déployés dans les 4 coins de la zone du réseau [Philipose et Rajesh, 2016] .

Les coordonnées des nœuds de référence sont (0,0), (0,50), (50,0 ), et (50,50). En outre, il est supposé que la communication entre les nœuds n'est possible que si les nœuds sont dans la portée du réseau.

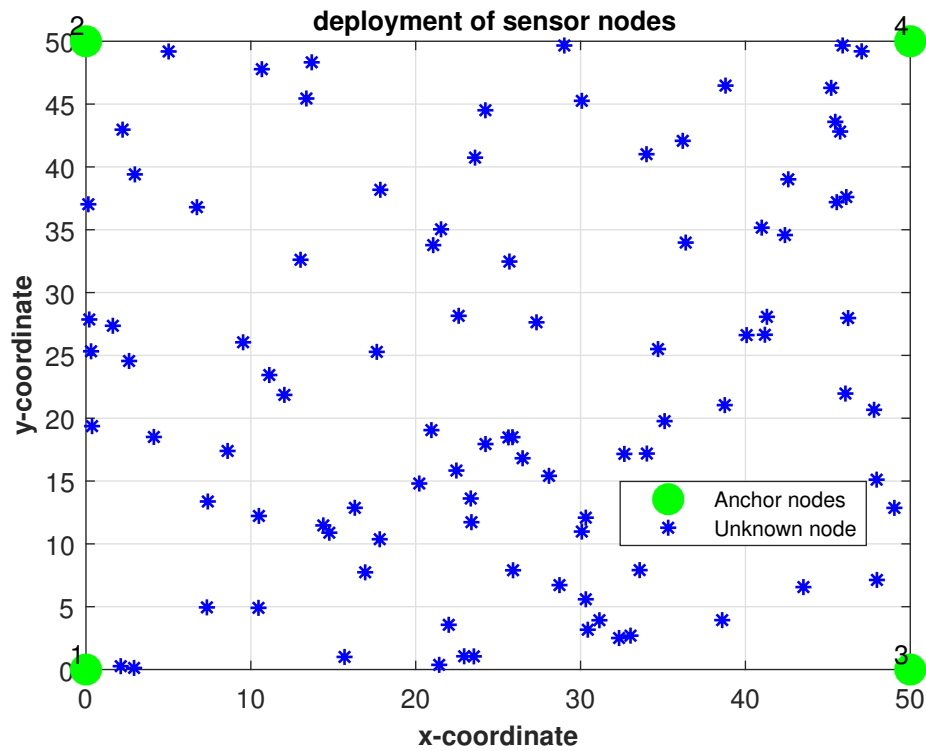


FIGURE 5.15 – Déploiement des nœuds dans le réseau.

La figure 5.16 présente le diagramme des relations de voisinage associé à la figure 5.15, dans lequel la portée de communication est fixée à 30 m. De plus, il est supposé que chaque nœud de capteur peut communiquer avec tous les autres nœuds de capteurs, la connectivité du réseau est illustré dans a figure 5.16. Le diagramme des relations de voisinage informe si les nœuds inconnus sont dans la portée (voisinage) des nœuds des références ou non.

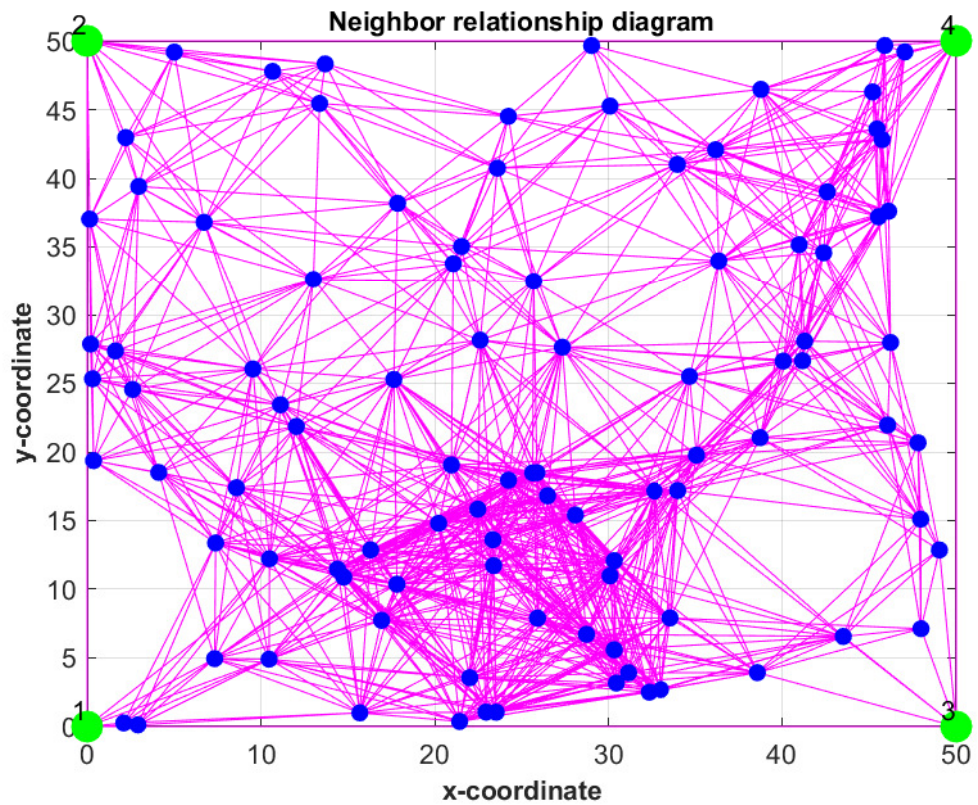


FIGURE 5.16 – La connectivité de communication entre les nœuds.

Dans des travaux antérieurs [Singh et al., 2013, Namin et Tinati, 2011, Li et Lei, 2009] il a été rapporté que l'algorithme PSO fournit des performances meilleures que l'algorithme de Recuit Simulé (RS), l'algorithme génétique (GA), les variantes de l'algorithme BBO et les autres méthodes d'optimisation mathématiques en matière de complexité de calcul et de précision de localisation.

Les figures 5.17 et 5.18 représentent les noeuds localisés par les algorithmes PSO et l'algorithme DE, respectivement.

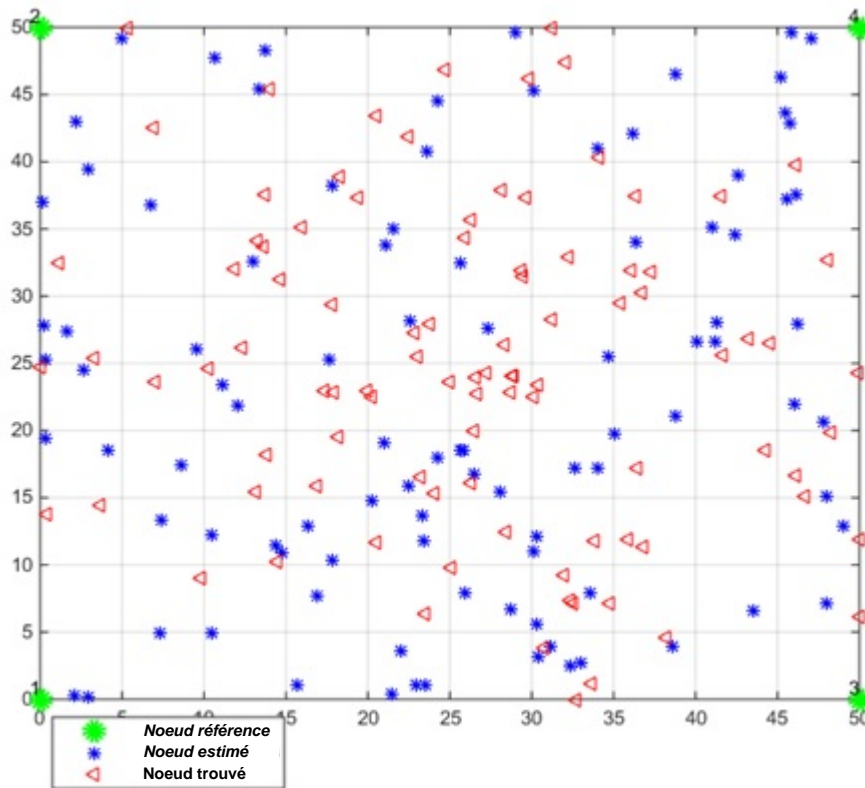


FIGURE 5.17 – Localisation des noeuds par l'algorithme de PSO .

En comparant les deux figures, on trouve que le taux des noeuds localisés par l'algorithme PSO est important mais la précision est moindre alors que les coordonnées des positions des noeuds localisés par l'algorithme DE sont importantes et sont plus précises.

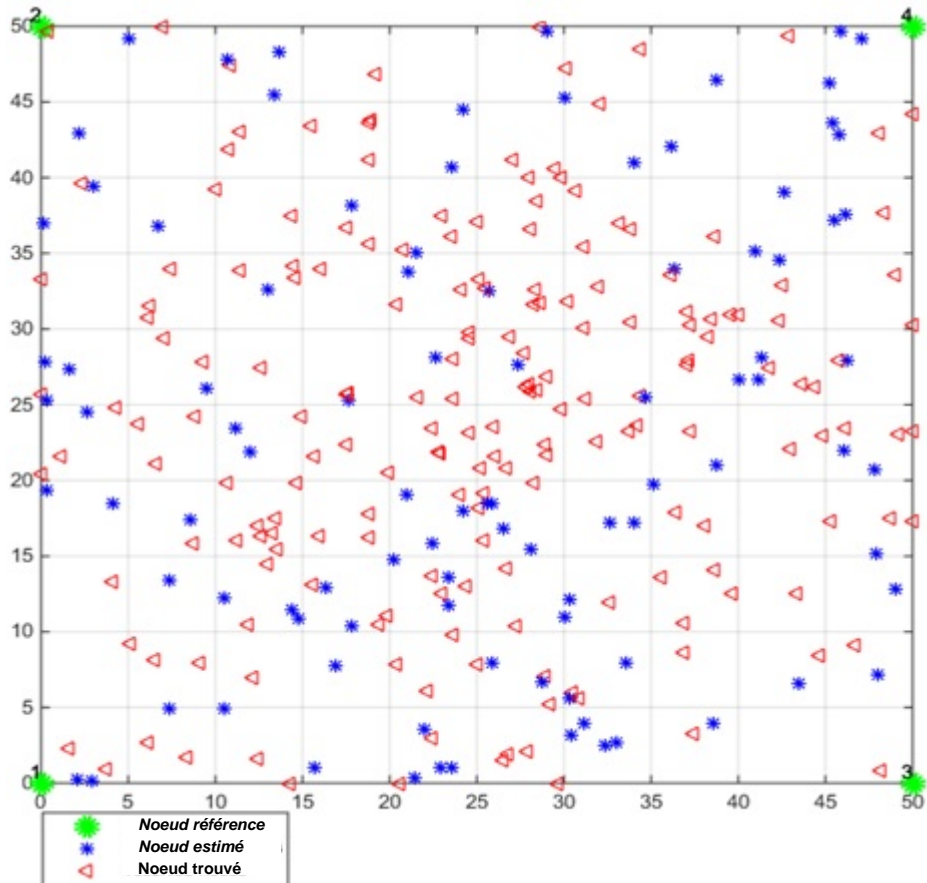


FIGURE 5.18 – Localisation des nœuds par l'algorithme DE .

La figure 5.19 représente les nœuds localisés par l'algorithme de chauve-souris, à partir de la figure 5.19, nous constatons que la majorité des nœuds localisés sont limités et concentrés légèrement dans le milieu du réseau, cela fait référence aux inconvénients de l'algorithme de chauve-souris (Bat), à cause que cet algorithme est faible dans la phase d'exploitation, malgré de sa puissance dans la phase l'exploration.

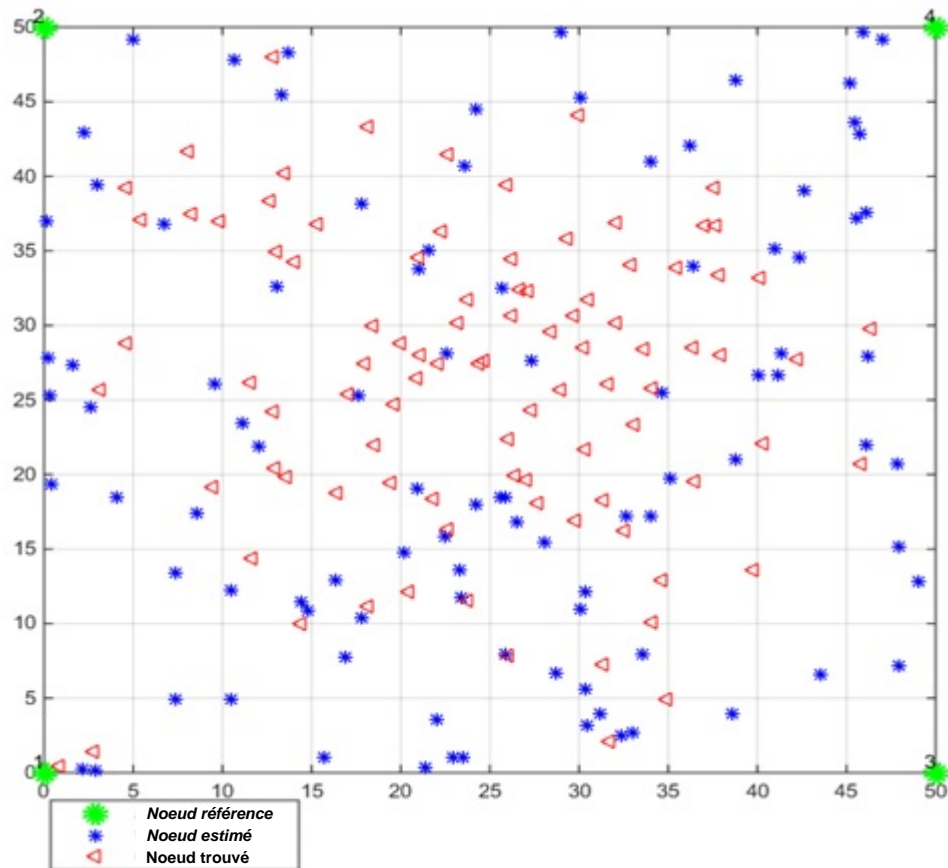


FIGURE 5.19 – Localisation des noeuds par l'algorithme de chauve-souris .

La figure 5.20 représente les noeuds localisés par l'algorithme de GWO, à partir de cette figure, nous constatons que la localisation des noeuds par l'algorithme GWO est focalisé dans la région de recherche variée dans l'intervalle (5-45 m), bien que cet algorithme (GWO) donne des bons résultats et efficace dans une zone étroite mais exécuter cet algorithme dans un champ de détection évolue rends l'algorithme faible devant la complexité du réseau, Par ailleurs, le temps de localisation basée sur l'algorithme de GWO est très important.

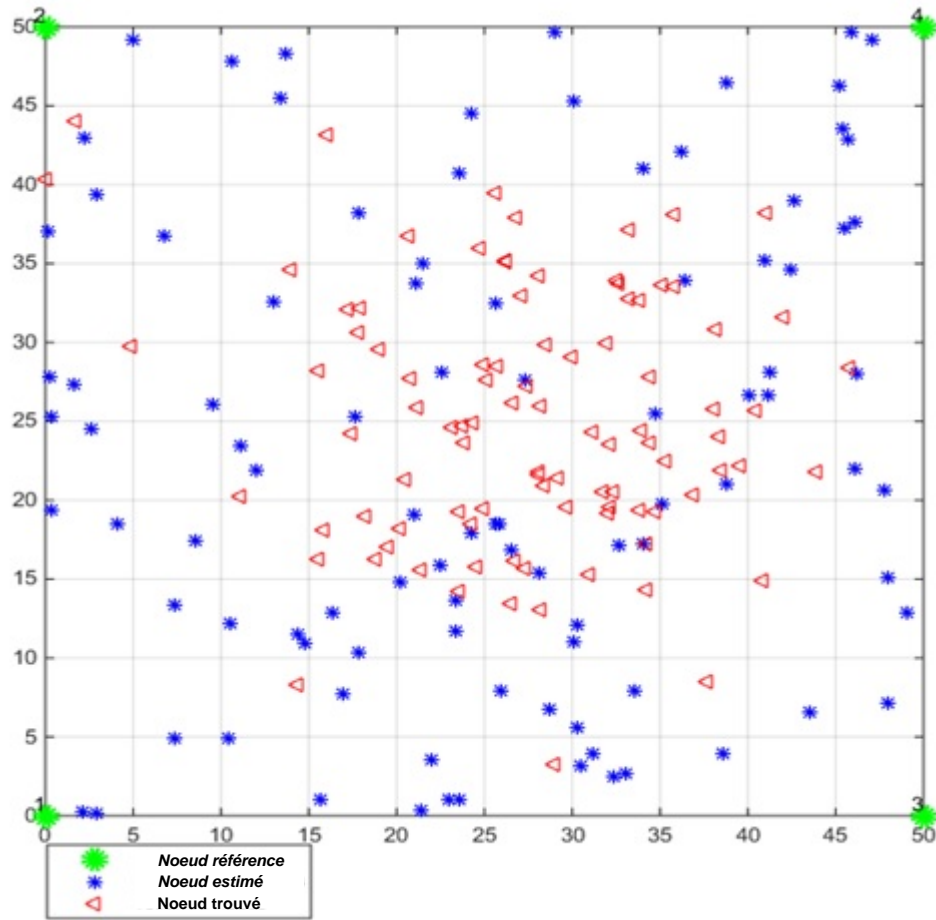


FIGURE 5.20 – Localisation des noeuds par l'algorithme de GWO .

L'ensemble des noeuds localisés par l'algorithme FPA sont présentés dans la figure 5.21. Bien que l'algorithme FPA ait prouvé son efficacité dans la localisation des noeuds. On analyse la figure 5.21, on trouve que l'algorithme FPA donne des résultats satisfaisants et les noeuds inconnus sont bien détectés. D'un autre côté, on trouve que l'algorithme FPA possède plusieurs paramètres à initialiser et à adapter pour chaque opération, ces paramètres rendent le temps de localisation basé sur cet algorithme très important.

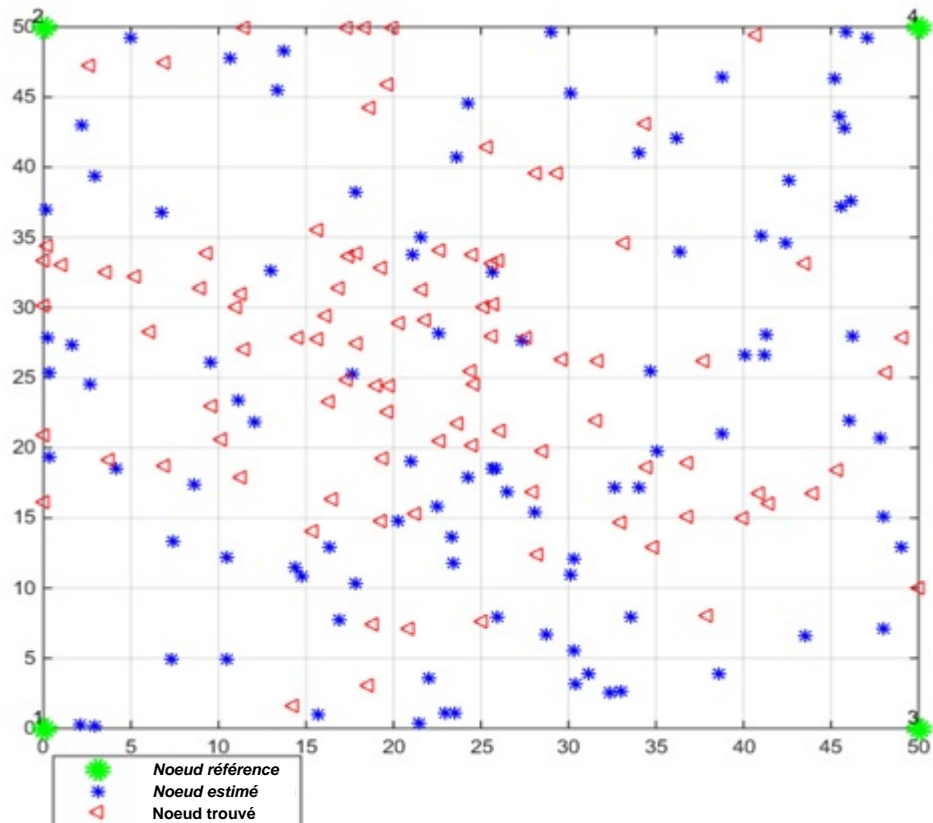


FIGURE 5.21 – Localisation des noeuds par l'algorithme de FPA .

La localisation basée sur l'algorithme MFOA est présentée dans la **figure 5.22**, par analyse du taux et la dispersion des noeuds localisés avec les autres algorithmes méta-heuristiques, on trouve que l'algorithme MFOA possède un déploiement meilleur et optimal où les zones des noeuds détectés sont importantes, par ailleurs, l'algorithme possède la capacité de supporter plus de densité des noeuds et résiste avec l'évolution du réseau.

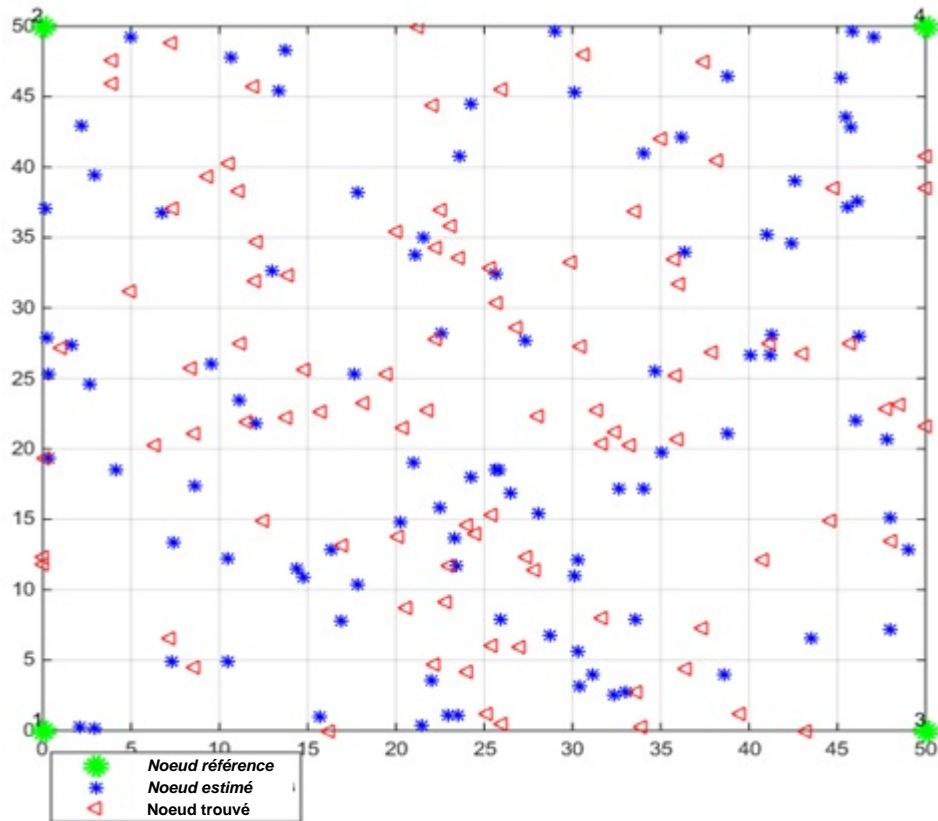


FIGURE 5.22 – Localisation des noeuds par l'algorithme de MFOA .

### 5.3.8 Résultat de localisation sous la portée de communication

L'impact de la portée de communication sur l'ensemble des algorithmes en matière d'erreur de localisation moyenne est présenté dans la figure 5.23. La variation de la portée de communication est entre 10 m à 40 m. à partir de la figure 5.23, on peut observer que l'erreur de localisation moyenne pour la plage de valeur de 5 est différente pour les algorithmes, cela revient aux paramètres d'initialisations qui sont aléatoires, par ailleurs, si la portée de communication est inférieure à 20 m, l'erreur de localisation moyenne est légèrement plus importante. Cela est dû au fait que le réseau n'est pas totalement connecté pour de nombreux noeuds. À mesure que la portée de communication augmente, l'erreur de localisation moyenne diminue considérablement. Cela est dû au fait qu'il y a plusieurs informations d'ancrages disponibles pour calculer l'emplacement des noeuds inconnus. Même ainsi, lorsque la plage de communication augmente jusqu'à une certaine valeur, l'erreur de localisation moyenne diminue légèrement. La précision de localisation dépend également de la densité de noeuds introduite. De plus, on peut observer que l'erreur de localisation moyenne diminue progressivement par rapport à l'augmentation de la densité des noeuds. Cela est dû au fait que lorsque la densité des noeuds augmente, la connectivité du réseau devient élevée et le nombre de noeuds disponibles dans la plage de communication augmente, ainsi, chaque noeud localisé à l'itération actuelle sera considéré comme un noeud de référence dans la prochaine itération. L'erreur de localisation ob-

tenue par les algorithmes DE et PSO est importante par rapport aux autres algorithmes, tandis que la convergence des algorithmes Bat, GWO et FPA est rapide, par rapport aux algorithmes DE et PSO, la convergence de MFOA est meilleure avec stabilité entre la portée de 20 et 25.

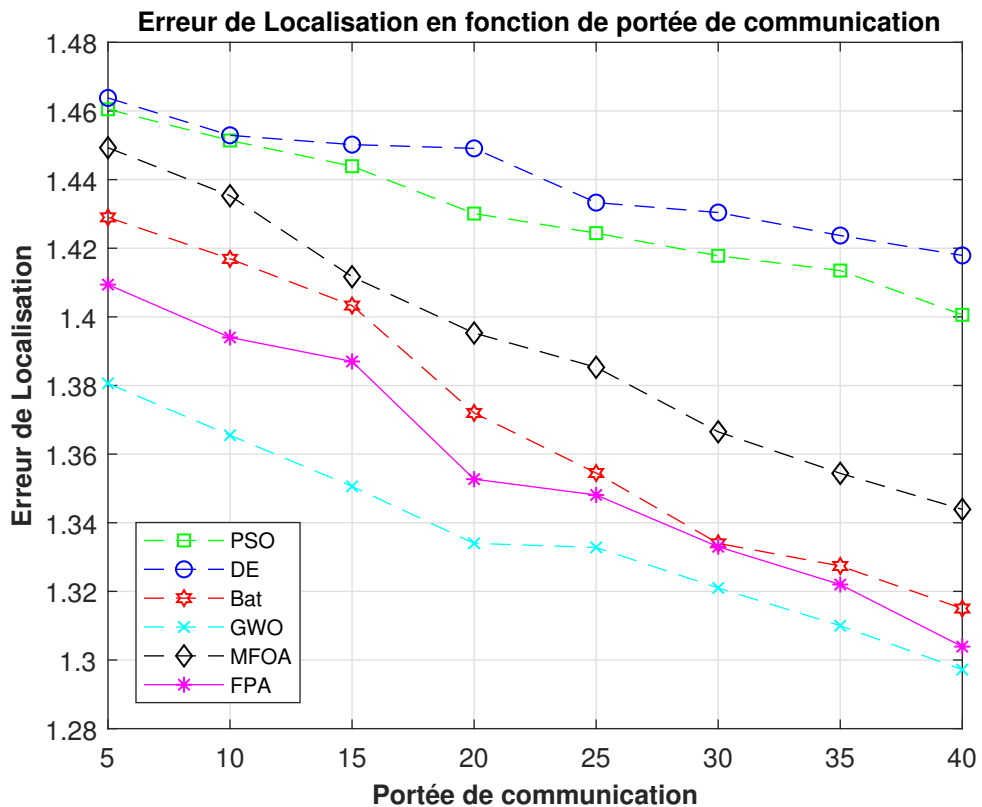


FIGURE 5.23 – Erreur de localisation vs plage de communication .

L'impact de la portée de communication sur l'ensemble des algorithmes en matière du temps de localisation moyenne est présenté dans la figure 5.24. A partir de la figure 5.24, on trouve que le temps de localisation obtenu par l'algorithme FPA est très important, pourtant elle a donné une erreur de localisation très satisfaisante, alors que le temps de localisation influence sur la consommation d'énergie des nœuds et donc sur la durée de vie du réseau. Dans l'autre côté, le temps de localisation obtenu par l'algorithme PSO est très faible car il a peu de paramètres à mettre à jour à chaque itération et le déploiement initial est aléatoire à chaque opération de localisation, de plus, la plage de communication faible donne une faible performance et les nœuds de références ne peuvent pas détecter des nœuds inconnus dans son environnement. la dernière partie du graphe, nous trouvons que le temps de localisation entre l'algorithme PSO et MFOA n'est pas important, tandis que, après la valeur 30 de la plage de communication, l'algorithme MFOA a dépassé les autres algorithmes et donne une valeur de localisation optimale. D'autre part, l'algorithme MFOA a obtenu une erreur très acceptable et une convergence par rapport à l'algorithme PSO et aux autres algorithmes. La convergence de l'algorithme MFOA est garantie car les mites ont toujours une tendance à mettre

à jour leurs positions par rapport aux flammes qui sont les solutions les plus prometteuses obtenues. Là encore, à partir de **la figure 5.23**, on trouve que l'erreur de localisation obtenue par les algorithmes DE et PSO est très importante et ils diminuent presque en même temps, sur **la figure 5.23**, l'algorithme MFOA a une erreur de localisation minimale que les autres algorithmes.

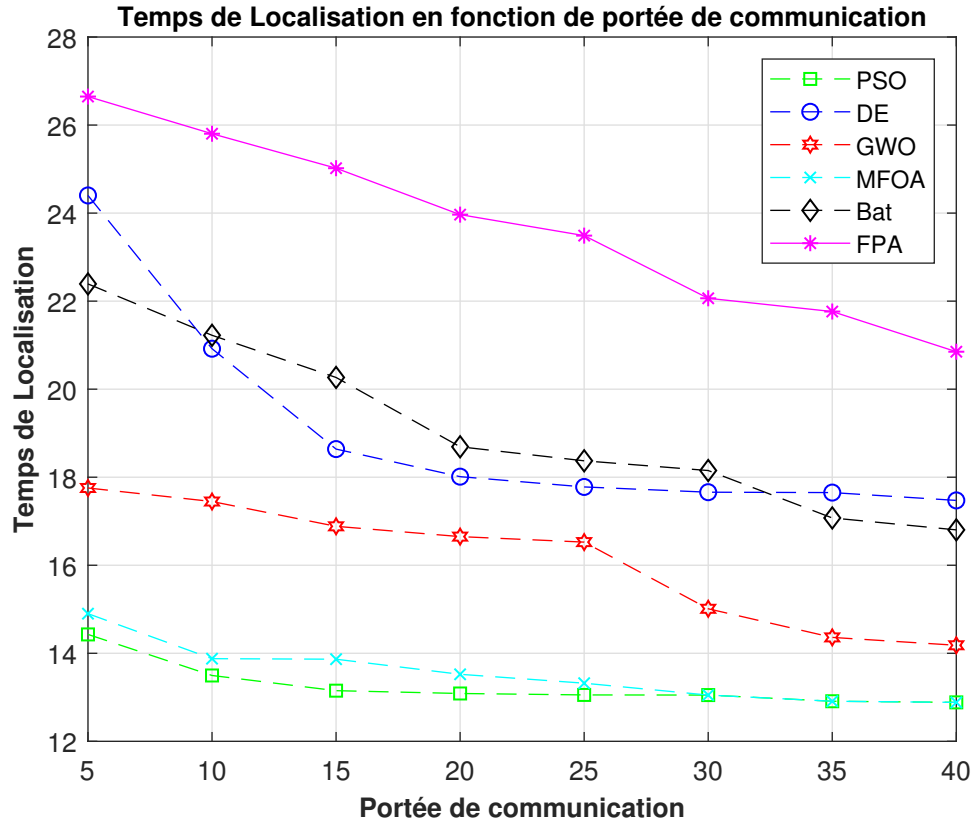


FIGURE 5.24 – Temps de localisation vs plage de communication .

### 5.3.9 Résultats de localisation sous le nombre d'itération

La figure 5.25 représente les valeurs de l'erreur de localisation par rapport au nombres d'itérations, à partir de la figure 5.25, on trouve que lorsque nous augmentons le nombre d'itérations, l'erreur de localisation diminue, cela est dû au fait que pour chaque itération, la population essaie de trouver la meilleure position pour chaque nœud, cela en raison de la possibilité de localiser un autre nœud amélioré lors du passage de l'itération  $i$  à l'itération  $i + 1$ . D'un autre côté, la valeur initiale de chaque algorithme est choisie de manière aléatoire de sorte que l'erreur de localisation initiale est différente. Mais après l'itération 10, l'erreur de localisation diminue pour chaque algorithme. D'après la figure 5.25, l'erreur de localisation obtenue par l'algorithme MFOA est meilleure et minimale que les autres algorithmes. Alors que l'algorithme GWO converge rapidement par rapport à l'algorithme FPA.

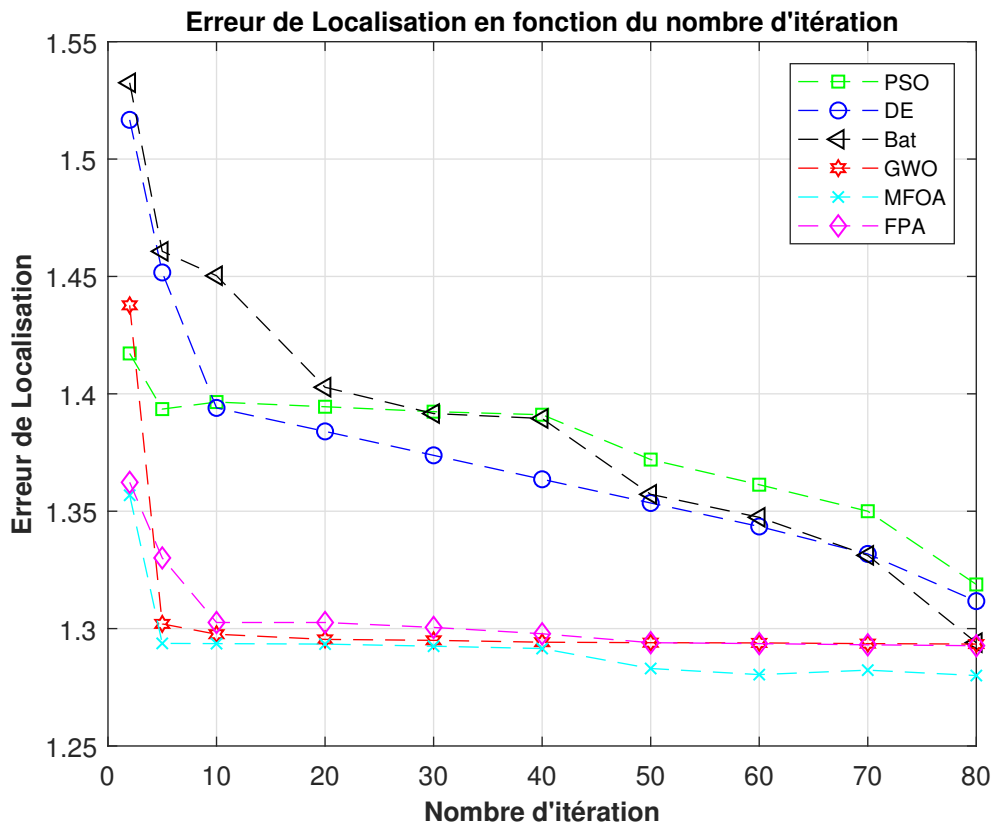


FIGURE 5.25 – Erreur de localisation vs nombre d'itérations .

L'impact du nombre d'itérations sur l'ensemble des algorithmes en termes du temps de localisation moyenne est illustré dans la figure 5.26. A partir de la figure 5.26, on trouve que, quand le nombre d'itérations augmente le temps de localisation augmente respectivement, cela est dû que les nœuds de références exploitent les populations pour retrouver d'autres solutions optimales. Alors que, le temps de localisation obtenu par l'algorithme FPA est considérable par rapport aux autres algorithmes, parce que l'algorithme FPA possède plusieurs paramètres et règles à adapter

pour chaque étape d'exécution et donc ce qui alourdit vraiment l'algorithme. en outre, L'algorithme de chauves-souris possède une convergence , ce qui permet de minimiser le temps de localisation, après la valeur d'itération 70 , elle obtient une valeur de localisation proche de celle-ci obtenu par l'algorithme de MFOA . L'algorithme de MFOA a obtenu un temps de localisation optimale par rapport aux autres algorithmes. Notant que lorsque le temps de localisation est petit, l'énergie consommée est conservée et minimale, et vice-versa. Le temps de localisation influe sur la consommation énergétique des nœuds et donc sur la durée de vie et la durabilité du réseau.

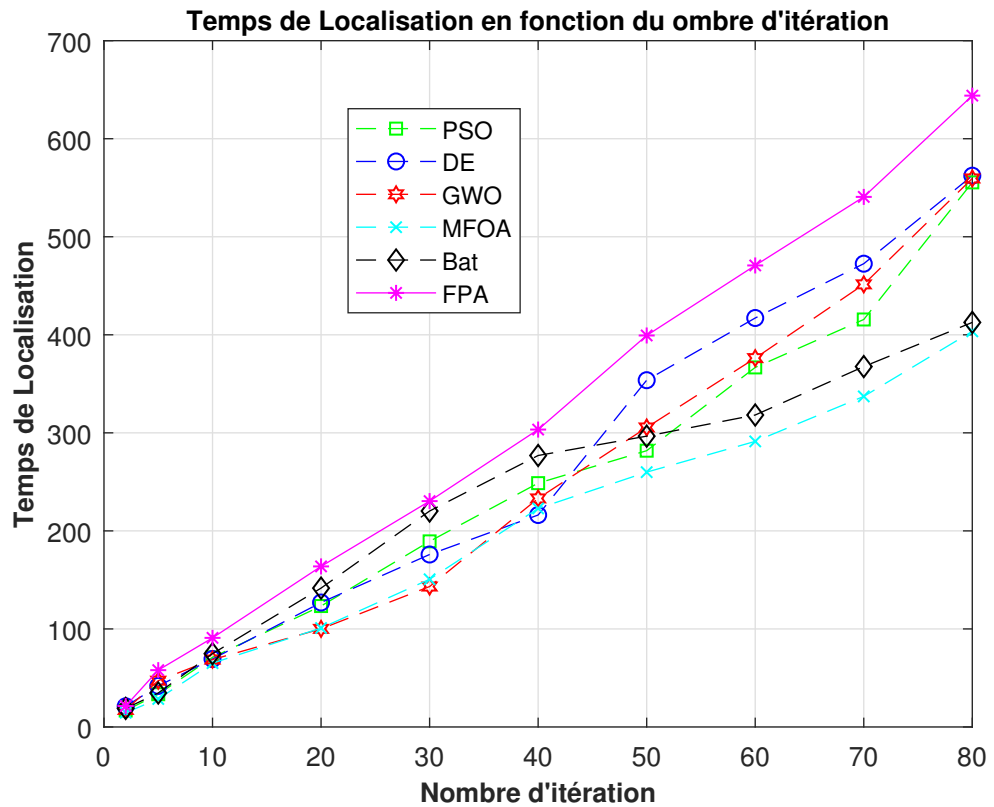


FIGURE 5.26 – Temps de localisation vs nombre d'itérations .

## 5.4 CONCLUSION

L'algorithme MFOA est un algorithme méta-heuristique très prometteur et intéressant qui a déjà été appliqué avec succès pour résoudre plusieurs problèmes d'optimisations [Muangkote et al., 2016] [Guvenc et al., 2017]. Ses avantages par rapport aux autres algorithmes d'optimisation sont la simplicité, la vitesse de recherche et l'hybridation simple avec d'autres algorithmes d'optimisations. Cependant, l'algorithme adapte la taille pour permettre à la population d'approcher rapidement la solution optimale globale, et l'adéquation de chaque solution est utilisée pour éviter la convergence locale. De plus, pour éviter la consommation d'énergie causée par une recherche insignifiante, l'approche restreint la population dans une certaine fourchette. Dans ce chapitre, l'algorithme d'optimisation de la flamme des mites est proposé pour résoudre le problème de localisation des nœuds dans les réseaux de capteurs sans fils. Cet algorithme est introduit dans un environnement distribué à l'aide d'une technique de connexion à sauts multiples et basée sur la plage. L'algorithme est proposé pour optimiser les performances de l'emplacement des nœuds. Des expériences approfondies ont été réalisées pour étudier l'impact de plusieurs facteurs comme la densité des nœuds de références, la variation de la taille de population et la plage de communication par l'évaluation de l'erreur de localisation moyenne et le temps de localisation moyenne. Cela fournit la base pour optimiser la localisation des nœuds en utilisant l'algorithme MFOA dans les applications des réseaux de capteurs sans fils pratiques. nous avons montré que l'algorithme MFOA est capable de prouver une exploration élevée et compétitive. De plus, les résultats obtenus prouvent que le MFOA équilibre correctement l'exploration et l'exploitation.

en outre, une étude comparative a été menée avec les algorithmes méta-heuristiques les plus populaires et le plus récent telle que : PSO, DE, Bat, GWO, FPA. L'étude comparative établie s'articule principalement sur les deux critères d'évaluation à optimiser : le temps de localisation moyenne et l'erreur de localisation moyenne, D'où l'évolution de l'étude est basée sur la plage de communication et le nombre d'itérations.

Les résultats expérimentaux obtenus ont prouvé l'efficacité et la puissance de l'algorithme MFOA. l'algorithme d'optimisation de la flamme des mites fonctionne mieux en matière de réduction de l'erreur de localisation moyenne et d'augmentation du taux de convergence, ce qui est favorable pour réduire la consommation de calcul et donc prolonger la durée de vie des nœuds de capteurs.

# MES CONTRIBUTIONS SCIENTIFIQUES

Notre thématique traite le problème de localisation des nœuds dans les réseaux de capteurs sans fil, La localisation est basée sur un nombre minimal de nœuds de références dans un environnement de surveillance distribuée, le schéma de localisation est introduite comme suit : premièrement, nous avons proposé une technique de localisation basée sur un algorithme métaheuristique appelé l'algorithme d'optimisation de la flamme des mites, le déploiement est effectué dans un environnement distribué, il s'agit d'une technique distribuée basée sur la portée de communication. Dans cette technique, nous classons les nœuds en trois types :

- . les nœuds de référence.
- . les nœuds cibles.
- . les nœuds inconnus. Les nœuds inconnus détectés dans la zone de captage devenu des nœuds cibles, les nœuds cibles se localisent à l'aide des nœuds de référence. Tout nœud cible localisé est considéré comme un nœud de référence et donc participe prochainement dans l'opération de localisation. Cette stratégie est appelée une stratégie itérative ou incrémentale ou multiniveau. Le schéma de localisation introduit est évalué sous deux critères de localisation :
  - . Le temps de localisation moyenne.
  - . L'erreur de localisation moyenne. Les résultats de simulation ont prouvé la convergence rapide de l'algorithme pour la localisation des nœuds. Les résultats obtenus prouvent que l'algorithme MFOA est très efficace et performant. Nous avons prouvé que le temps de localisation et l'erreur de localisation sont optimaux dans le schéma proposé basé sur cet algorithme par apport aux autres algorithmes métaheuristicques.

**Une version améliorée de l'algorithme de chauve-souris** a été formulé en fonction de la fréquence changeante de l'événement périodique lorsqu'un observateur se déplace par rapport à sa source [Mihoubi et al., 2018]. Ce changement est connu sous le nom d'effet Doppler. La fréquence changeante permet l'algorithme convergé rapidement pendant le déplacement des chauves-souris et améliore le taux de localisation des solutions optimales et donc minimise l'erreur de localisation. **Algorithme d'optimisation des mouches des fruits (EFOA) améliorée** est proposée pour gérer la localisation dans les réseaux de capteurs sans fils [Mihoubi et al., 2020]. En outre, l'optimisation multicritères est traitée erreur de localisation et le temps de convergence. Le résultat de la simulation est exécuté en deux parties, la première partie, les paramètres de l'algorithme proposé sont évalués sous des scénarios variants, dans la deuxième partie, une étude comparative est menée avec d'autres algorithmes d'optimisations.

Le résultat obtenu montre que l'algorithme EFOA donne une erreur de localisation optimale et une convergence rapide.

# CONCLUSION GÉNÉRALE

La localisation des noeuds dans les réseaux de capteurs sans fils a reçu une attention croissante au cours de la dernière décennie. Il fournit non seulement la position géographique d'un nœud de capteur, mais remplit également les conditions requises pour le routage géographique, l'interrogation spatiale et la diffusion des données. La recherche dans la localisation des nœuds des réseaux de capteurs sans fils a connu un progrès significatif, un certain nombre d'algorithmes efficaces ont été proposés, mais la stabilité n'est pas encore atteinte. Cela est dû aux faibles ressources (stockage, batterie, processeur) et aux environnements de déploiement difficiles. Actuellement, aucune des techniques de localisation n'est capable de remplir toutes ces contraintes de satisfaction à la fois. La plupart des algorithmes de localisation existant pour les réseaux de capteurs sans fils statique ont été conçus pour fonctionner avec au moins trois nœuds d'ancrage, sauf dans les cas où l'antenne directionnelle est utilisée. L'utilisation de l'antenne augmente non seulement le coût, mais également la taille du nœud ainsi que la complexité de l'algorithme. À mesure que le nombre des nœuds des références requis dans un réseau augmente, le coût global du réseau augmente également. De plus, la consommation énergétique du réseau augmente, mais le temps de localisation de l'ensemble du réseau diminue. En outre, les nœuds de références installées avec le GPS ne fonctionnent pas parfaitement. Par conséquent, nous avons actuellement besoin d'une nouvelle technique qui résoudra les problèmes suivants :

- . Réduire le nombre de nœuds de références requis.
- . Localiser les nœuds de capteurs dans les zones où le GPS ne fonctionne pas parfaitement.
- . Minimiser l'erreur de localisation.
- . Minimiser le temps de localisation.

Dans cette thèse, nous avons proposé une technique de localisation pour les réseaux de capteurs sans fils basé sur une métaheuristique récente appelée l'algorithme d'optimisation de la flamme des mites.

La deuxième partie de la thèse décrit l'historique du réseau de capteurs sans fil , ainsi l'évolution dans les différents domaines d'application, ensuite , on a détaillé son architecture ses fondements, les challenges qui ont affronté depuis leur apparition et à travers le développement accéléré des autres techniques ,Ainsi, ses domaines d'application et d'évolution du réseau de capteurs sans fil avec Internet des Objets ( IoT) dans certains domaines d'application. La troisième partie de la thèse décrit le problème localisation des nœuds dans les réseaux de capteurs sans fils , l'évolution de la problématique est discuté en se basant sur les approches classiques proposées en premier lieu, par la suite , les approches actuelles sont analysées dans les différents paradigmes ,En outre , une étude d'évaluation comparative a été introduit, avec cela, les phases principales de

la localisation sont discutées, analysées et évaluées, et finalement, la tendance et la solution actuelle proposée pour traiter le problème de localisation des nœuds dans les réseaux de capteurs sans fils. La quatrième partie de la thèse présente en détail une étude complète de l'algorithme d'optimisation appelé l'algorithme d'optimisation de la flamme des mites (MFOA) et analyse ses caractéristiques principales. L'algorithme MFOA est considéré comme l'un des algorithmes méta-heuristiques prometteurs et appliqué avec succès dans divers problèmes d'optimisations dans un large éventail de domaines. On a décrit la littérature disponible sur l'algorithme MFOA, y compris ses variantes et ses phases d'optimisations, ses paramètres, son évolution et hybridation, l'analyse théorique et les comparaisons de MFOA avec d'autres algorithmes. Les conclusions se sont concentré sur les travaux en cours sur l'algorithme MFOA, soulignent ses faiblesses et suggèrent de futures directions de recherche possibles. La cinquième partie de la thèse est dédié pour éclaircir et définir la plateforme de simulation pour élaborer nos travaux de recherche MATLAB, ainsi, l'organigramme du processus de localisation basé sur l'algorithme MFOA. Avec cela, des expériences approfondies ont été réalisé pour étudier les impacts de plusieurs facteurs du réseau comme la densité d'ancrage, la densité des nœuds et la plage de communication sur l'algorithme d'optimisation de la flamme des mites, deux métriques fondamentales ont été évalué : l'erreur de localisation moyenne et le temps demandé pour la localisation. Nous avons démontré que l'algorithme MFOA est capable de montrer une exploration élevée et compétitive. De plus, les résultats obtenus prouvent que l'algorithme MFOA équilibre correctement l'exploration et l'exploitation. De plus, une étude comparative a été menée avec les algorithmes métaheuristiques les plus populaires et le plus récent telle que : PSO, DE, Bat, GWO, FPA.

L'étude comparative établie s'articule principalement sur les deux critères d'évaluation à optimiser : le temps de localisation moyenne et l'erreur de localisation moyenne, D'où l'évolution de l'étude est basée sur la plage de communication et le nombre d'itérations. Les résultats expérimentaux obtenus ont prouvé l'efficacité et la puissance de l'algorithme de MFOA. L'algorithme d'optimisation de la flamme des mites fonctionne mieux en matière de réduction de l'erreur de localisation moyenne et d'augmentation du taux de convergence, ce qui est favorable pour réduire la consommation de calcul et donc prolonger la durée de vie des nœuds de capteurs. La dernière partie de la thèse décrit une conclusion générale qui englobe notre thèse de recherche, une discussion est introduite sur le future travail et les perspectives proposées pour améliorer les performances de localisation dans les réseaux de capteurs sans fils.

## PERSPECTIVES

La plupart du temps, une solution à un problème donne beaucoup de problèmes à étudier, en outre, le problème de localisation dans les réseaux de capteurs sans fils n'est pas encore entièrement résolu. Dans ce qui suit, nous énumérons les travaux possibles qui nécessitent une attention particulière et visent à émerger des travaux menés dans cette thèse :

1. La précision de la localisation est principalement affectée par les tech-

niques de communication utilisées. Chaque technique de télémétrie est gravement affectée par le comportement du canal sans fil dans différents environnements, par conséquent, pour une localisation précise, des problèmes tels que l'évanouissement du signal, les trajets multiples, le bruit additif, etc. doivent être traités.

2. L'erreur de mesure de la distance entre les nœuds doit être traitée avec un étalonnage approprié car la plupart des algorithmes de localisation dépendent de la distance par paires.
3. En outre, la localisation dans certains environnements spécifiques comme les environnements sous-marins n'a pas été beaucoup explorée.
4. Dans l'algorithme de localisation proposé, il existe une contrainte stricte sur la connectivité entre les nœuds. Une extension naturelle du travail est d'assouplir les conditions de connectivité entre les nœuds.
5. Nous modifierons l'algorithme de localisation pour qu'il soit applicable à une topologie en évolution rapide. Dans la forme actuelle, une topologie en évolution rapide doit déclencher un changement rapide des messages entre les nœuds et la station de base, ce qui pourrait créer une congestion dans le réseau et éventuellement une perte des informations transférées.
6. la complexité du calcul de l'algorithme de localisation est considérable d'où la complexité de calcul de l'algorithme MFOA dépend du nombre de mites, du nombre de variables, du nombre maximal d'itérations et du mécanisme de tri des flamme à chaque instant  $t$ . Vu nous utilisons l'algorithme de tri rapide, le tri de  $O(n * \log(n))$  et de  $O(n^2)$  se fait respectivement dans le meilleur et le pire des cas. Compte tenu de la fonction  $P$ , par conséquent, la complexité globale du calcul est défini comme suit :

$$O(MFOA) = O(t(O(tri_{rapide}) + O(misejourdelaposition))) \quad (5.1)$$

$$O(MFOA) = O(t(n^2 + n * 2)) = O(t(n^2 + t * n * d)) \quad (5.2)$$

où  $n$  est le nombre de mites,  $t$  est le nombre maximum d'itérations et  $d$  est le nombre de variables. donc nous allons travailler pour réduire encore la complexité de calcul. Cela signifie que nous allons faire évoluer la stratégie de positionnement pour réduire le nombre d'échanges de messages entre les nœuds.

7. L'extension du schéma en trois dimensions est aussi intéressante et principalement dans un environnement intérieur.
8. La consommation d'énergie est un autre critère décisif et principal qui sera étudié pour tester la résistance et la durabilité du réseau. Dans ce but, les travaux futurs doivent se concentrer sur l'optimisation de l'énergie dans l'application en temps réel du RCSF. Par conséquent, la localisation des nœuds grâce à l'optimisation multicritère et l'évaluation de la consommation d'énergie rendent l'algorithme proposé fructueux et donnent de meilleures performances au cours de l'évolution du réseau.

Le réseau de capteurs sans fil sous-marin est un autre domaine actif peut-être abordé pour les problèmes de localisation des nœuds. Ce domaine est très délicat spécialement dans la surveillance.

En matière d'optimisation, plusieurs techniques peuvent être introduites :

- 1- Dans [Blum et Roli, 2008] [Ting et al., 2015], il existe plusieurs stratégies d'hybridation qui peuvent être appliquées avec l'algorithme de chauve-souris pour améliorer ses performances, à titre d'exemple, l'algorithme de chauve-souris et l'algorithme de MFOA pourraient être combinés avec une réduction relative comme fonction de fitness pour obtenir de meilleures performances. Si une réduction relative est utilisée pour la sélection d'entités, le temps de calcul du modèle de sélection d'entité augmente. Par conséquent, cela nécessite une intégration avec d'autres techniques de calcul, tel que le calcul parallèle, pour réduire le temps de traitement écoulé. En outre, renforcer l'algorithme MFOA est avantageuse, le réglage des paramètres est plutôt statique. Nous recommandons que les méthodes proposées de l'algorithme MFOA soient intégrées à une autre méthode pour régler de manière flexible les paramètres associés et les rendre mutables ou dynamiques.
- 2- Introduire des techniques d'apprentissage approfondies pour traiter ce type de problème d'optimisation, avec prendre en considération la consommation d'énergie, vu ces techniques demandent beaucoup de calcul et demandent des machines de hautes performances [Al-Obeidat et al., 2019] [Talbi, 2020].
- 3- Une stratégie autre importante est d'intégrer des récentes techniques d'apprentissage approfondies pour améliorer l'optimisation des algorithmes de métaheuristique, comme il peut être déployé dans un environnement des nœuds de capteurs mobiles.

# BIBLIOGRAPHIE

- Karati A et Biswas GP. Provably secure and authenticated data sharing protocol for iot based crowdsensing network. *rans Emerg Telecommun Technol*, 30(4) :1–22, 2019.
- S.M. Abd-Elazim et E.S. Ali. A hybrid particle swarm optimization and bacterial foraging for optimal power system stabilizers design. *International Journal of Electrical Power and Energy Systems*, 46 :334–341, 2013.
- Fereydoon Abdi et Abolfazl Toroghi Haghighat. A hybrid rssi based localization algorithm for wsn using a mobile anchor node. *Fifth International Conference on Computing, Communications and Networking Technologies (ICCCNT)*, pages 1–6, 2014.
- M. A. Adnan, M. Abdur Razzaque, I. Ahmed, et I. F. Isnin. Bio-mimic optimization strategies in wireless sensor networks : a survey. *Sensors*, 1(14) :299–345, 2014.
- T. Ahmad, X. J. Li, et B. Seet. Fuzzy-logic based localization for mobile sensor networks\*. Dans *2019 2nd International Conference on Communication, Computing and Digital systems (C-CODE)*, pages 43–47, 2019.
- T. Ahmad, X. J. Li, et B. C. Seet. Parametric loop division for 3d localization in wireless sensor networks. *Sensors (Basel, Switzerland)*, 17(7) :1697, 2017.
- B. Akay et D. Karaboga. Artificial bee colony algorithm for large-scale problems and engineering design optimization. *Journal of Intelligent Manufacturing*, 23 :1001–1014, 2012.
- Feras Al-Obeidat, Nabil Belacel, et Bruce Spencer. Combining machine learning and metaheuristics algorithms for classification method proaftn. Dans *Enhanced Living Environments*, 2019.
- L. K. Alazzawi, A. M. Elkateeb, A. Ramesh, et W. Aljuhar. Scalability analysis for wireless sensor networks routing protocols. Dans *22nd International Conference on Advanced Information Networking and Applications - Workshops (aina workshops 2008)*, pages 139–144, 2008.
- M. Aldwaik et H. Adeli. Advances in optimization of highrise building structures. *Structural and Multidisciplinary Optimization*, 50 :899–919, 2014.
- I. Amundson et X. Koutsoukos. A survey on localization for mobile wireless sensor networks. Dans *MELT*, 2009.

- W. Anani, A. Ouda, et A. Hamou. A survey of wireless communications for iot echo-systems. Dans *2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*, pages 1–6, 2019.
- Yunus Emre Aslan, Ibrahim Korpeoglu, et Özgür Ulusoy. A framework for use of wireless sensor networks in forest fire detection and monitoring. *Computers, Environment and Urban Systems*, 36(6) :614–625, 2012.
- H. Ayadi, A. Zouinkhi, T. Val, A. van den Bossche, et M. N. Abdelkrim. Network lifetime management in wireless sensor networks. *IEEE Sensors Journal*, 18(15) :6438–6445, 2018.
- M. Ayaz, M. Ammad-uddin, I. Baig, et e. M. Aggoune. Wireless sensor’s civil applications, prototypes, and future integration possibilities : A review. *IEEE Sensors Journal*, 18(1) :4–30, 2018.
- N. A. B. A. Aziz, A. W. Mohemmed, et B. S. D. Sagar. Particle swarm optimization and voronoi diagram for wireless sensor networks coverage optimization. Dans *2007 International Conference on Intelligent and Advanced Systems*, pages 961–965, 2007.
- Peng Sun Azzedine Boukerche. Connectivity and coverage based protocols for wireless sensor networks. *Ad Hoc Networks*, 80 :54–69, 2018.
- J. Bachrach et C. Taylor. Localization in sensor networks. Dans *Handbook of Sensor Networks : Algorithms and Architectures*, Hoboken ,John Wiley and Sons, 2005.
- Shonak Bansal. Performance comparison of five metaheuristic nature-inspired algorithms to find near-optimal solutions for wdm systems. *Artificial Intelligence Review*, pages 1 – 47, 2020.
- Abdulaziz Y. Barnawi, Gamel A. Mohsen, et Essa Q. Shakra. Performance analysis of rpl protocol for data gathering applications in wireless sensor networks. *Procedia Computer Science*, 151 :185–193, 2019.
- S. Bera, S. Misra, et J. J. P. C. Rodrigues. Cloud computing applications for smart grid : A survey. *IEEE Transactions on Parallel and Distributed Systems*, 26(5) :1477–1494, 2015.
- Dimitris Bertsimas et John Tsitsiklis. Simulated Annealing. *Statistical Science*, 8 :10 – 15, 1993.
- M. Birattari, L. Paquete, T. Stützle, et Klaus Varrentrapp. Classification of metaheuristics and design of experiments for the analysis of components. Dans *Wiley*, 2001.
- Pratik Biswas, TzuChen Lian, TaChung Wang, et Yinyu Ye. Semidefinite programming based algorithms for sensor network localization. *ACM Transactions on Sensor Networks*, 2(2) :188–220, 2006.
- C. Blum et A. Roli. Metaheuristics in combinatorial optimization : Overview and conceptual comparison. *ACM Comput. Surv.*, 35 :268–308, 2003a.

- C. Blum et A. Roli. Metaheuristics in combinatorial optimization : Overview and conceptual comparison. *ACM Computing Surveys*, 35(3) : 268–308, 2003b.
- Christian Blum et Andrea Roli. Hybrid metaheuristics : An introduction. Dans *Hybrid Metaheuristics*, 2008.
- [ Z. Bojkovic et B. Bakmaz. survey on wireless sensor networks deployment. *WSEAS TRANSACTIONS on COMMUNICATIONS*, 7(12) : 1172–1181, 2008.
- A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura, et A. A. F. Loureiro. Localization systems for wireless sensor networks. *IEEE Wireless Communications*, 14(6) :6–12, 2007.
- E. Burke, M. Gendreau, M. Hyde, G. Kendall, G. Ochoa, E. Özcan, et R. Qu. Hyper-heuristics : a survey of the state of the art. *Journal of the Operational Research Society*, 64 :1695–1724, 2013.
- I. Butun, S. D. Morgera, et R. Sankar. A survey of intrusion detection systems in wireless sensor networks. *IEEE Communications Surveys Tutorials*, 16(1) :266–282, 2014.
- C.Eder, V. Valente, N. Donaldson, et A. Demosthenous. A cmos smart temperature and humidity sensor with combined readout. *Sensors (Basel, Switzerland)*, 14(9) :17192–17211, 2014.
- Chee-Yee Chong et S. P. Kumar. Sensor networks : evolution, opportunities, and challenges. *Proceedings of the IEEE*, 91(8) :1247–1256, 2003.
- I. Ciornei et E. Kyriakides. Hybrid ant colony-genetic algorithm (gaapi) for global continuous optimization. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42 :234–245, 2012.
- Carlos A. Coello Coello. Use of a self-adaptive penalty approach for engineering optimization problems. *Computers in Industry*, 41 :113–127, 2000.
- Nam CS, Han YS, et Shin DR. Multi-hop routing-based optimization of the number of cluster-heads in wireless sensor networks. *Sensors (Basel)*, 11(3) :2875–2884, 2011.
- P. T. Daely et S. Y. Shin. Range based wireless node localization using dragonfly algorithm. Dans *2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN)*, pages 1012–1015, 2016.
- N. T. Dinh et Y. Kim. Auto-configuration in wireless sensor networks : A review. *Sensors (Basel, Switzerland)*, 19(19) :4281, 2019.
- L. Doherty, K. S. J. pister, et L. El Ghaoui. Convex position estimation in wireless sensor networks. Dans *Proceedings IEEE INFOCOM 2001. Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Society (Cat. No.01CH37213)*, volume 3, pages 1655–1663 vol.3, 2001.

- M. Dorigo et M. Birattari. Swarm intelligence. *Scholarpedia*, 2 :1462, 2007.
- Haifeng Du, X. Wu, et Jian Zhuang. Small-world optimization algorithm for function optimization. Dans *ICNC*, 2006.
- A. Dâmaso, N. Rosa, et P. Maciel. Reliability of wireless sensor networks. *Sensors (Basel, Switzerland)*, 14(9) :15760–15785, 2014.
- A. Engelbrecht. Fundamentals of computational swarm intelligence. Dans *John Wiley and Sons, Inc*, 2005.
- H. Eskandar, Ali Sadollah, A. Bahreininejad, et M. Hamdi. Water cycle algorithm - a novel metaheuristic optimization method for solving constrained engineering optimization problems. *Computers and Structures*, 110 :151–166, 2012.
- Reza Faturechi et Elise Miller-Hooks. A mathematical framework for quantifying and optimizing protective actions for civil infrastructure systems. *Comput. Aided Civ. Infrastructure Eng.*, 29 :572–589, 2014.
- L. Fogel, A. J. Owens, et M. J. Walsh. Artificial intelligence through simulated evolution. Dans *Evolutionary Computation : The Fossil Record*, 1966.
- R. Formato. Central force optimization : A new meta-heuristic with applications in applied electromagnetics. *Progress in Electromagnetics Research-pier*, 77 :0–0, 2007.
- K. D. Frank, C. Rich, et T. Longcore. Effects of artificial night lighting on moths. Dans *Ecological consequences of artificial night lighting*, pages 305–344, 2006.
- H. Gao et X. Zhang. A markov-based road maintenance optimization model considering user costs. *Comput. Aided Civ. Infrastructure Eng.*, 28 : 451–464, 2013.
- Kevin J. Gaston, Jonathan Bennie, Thomas W. Davies, et John Hopkins. The ecological impacts of nighttime light pollution : a mechanistic appraisal. *Biological reviews of the Cambridge Philosophical Society*, 88 4 : 912–27, 2013.
- D. S. Ghataoura, J. E. Mitchell, et G. E. Matich. Networking and application interface technology for wireless sensor network surveillance and monitoring. *IEEE Communications Magazine*, 49(10) :90–97, 2011.
- M. GomesPereira de Lacerda, H. Deandrade Amorim Neto, T. Bernardaludermir, H. Kuchen, et F. Buarquede Lima Neto. Population size control for efficiency and efficacy optimization in population based metaheuristics. Dans *2018 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8, 2018.
- A. Gopakumar et L. Jacob. Localization in wireless sensor networks using particle swarm optimization. Dans *2008 IET International Conference on Wireless, Mobile and Multimedia Networks*, pages 227–230, 2008.

- S. Goyal et M. S. Patterh. Flower pollination algorithm based localization of wireless sensor network. Dans *2015 2nd International Conference on Recent Advances in Engineering Computational Sciences (RAECS)*, pages 1–5, 2015.
- Sonia Goyal et Manjeet Singh Patterh. Wireless sensor network localization based on bat algorithm. *Int J Emerg Technol Comput Appl Sci (IJETCAS)*, 5 :507–512, 2013.
- F. Granda, L. Azpilicueta, C. Vargas-Rosales, P. Lopez-Iturri, E. Aguirre, et F. Falcone. Integration of wireless sensor networks in intelligent transportation systems within smart city context. Dans *2018 IEEE International Symposium on Antennas and Propagation USNC/URSI National Radio Science Meeting*, pages 375–376, 2018.
- V. Granville, M. Krivánek, et J. Rasson. Simulated annealing : A proof of convergence. *IEEE Trans. Pattern Anal. Mach. Intell.*, 16 :652–656, 1994.
- Q. Guan, F. Ji, Y. Liu, H. Yu, et W. Chen. Distance-vector-based opportunistic routing for underwater acoustic sensor networks. *IEEE Internet of Things Journal*, 6(2) :3831–3839, 2019.
- Q. Guo, Y. Zhang, J. Lloret, B. Kantarci, et W. K. G. Seah. A localization method avoiding flip ambiguities for micro-uavs with bounded distance measurement errors. *IEEE Transactions on Mobile Computing*, 18(8) :1718–1730, 2019.
- Govind P. Gupta et Sonu Jha. Integrated clustering and routing protocol for wireless sensor networks using cuckoo and harmony search based metaheuristic techniques. *Engineering Applications of Artificial Intelligence*, 68 :101–109, 2018.
- U. Guvenc, S. Duman, et Y. Hinislioglu. Chaotic moth swarm algorithm. Dans *2017 IEEE International Conference on INnovations in Intelligent Systems and Applications (INISTA)*, pages 90–95, 2017.
- Kun Hao, Haifeng Shen, Yonglei Liu, Beibei Wang, et Xiujuan Du. Integrating localization and energy-awareness : A novel geographic routing protocol for underwater wireless sensor networks. *Mobile Networks and Applications*, 23 :1427–1435, 2018.
- R. Harikrishnan, V. J. S. Kumar, et P. S. Ponmalar. Differential evolution approach for localization in wireless sensor networks. Dans *2014 IEEE International Conference on Computational Intelligence and Computing Research*, pages 1–4, 2014.
- A. Hatamlou. Black hole : A new heuristic optimization approach for data clustering. *Inf. Sci.*, 222 :175–184, 2013.
- Tian He, Chengdu Huang, Brain M. Blum, John A. Stankovic, et Tarek Abdelzaher. Range-free localization schemes for large scale sensor networks. Dans *MobiCom '03 : Proceedings of the 9th annual international conference on Mobile computing and networking September, 2003*, page 81–95, 2003.

- W. R. Heinzelman, A. Chandrakasan, et H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. Dans *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, pages 10 pp. vol.2-, 2000.
- J. Holland et J. Reitman. Cognitive systems based on adaptive algorithms. *SIGART Newsl.*, 63 :49, 1977.
- John H. Holland. Genetic algorithms. *Scientific American*, 267 1 :66–73, 1992.
- J. Holler, V. Tsiatsis, C. Mulligan, S. Karnouskos, S. Avesand, et D. Boyle. Introduction to a new age of intelligence ,amsterdam, the netherlands : Elsevier, 2014. Dans *From Machine-to-Machine to the Internet of Things :*, page 352, 2014.
- Hongyang Chen, Kaoru Sezaki, Ping Deng, et Hing Cheung So. An improved dv-hop localization algorithm for wireless sensor networks. Dans *2008 3rd IEEE Conference on Industrial Electronics and Applications*, pages 1557–1561, 2008.
- Binh HTT, Hanh NT, et Dey N. Improved cuckoo search and chaotic flower pollination optimization algorithm for maximizing area coverage in wireless sensor networks. *Neural Comput Appl*, 30(7) :2305–2317, 2018.
- Rechenberg I. Evolutionsstrategien. *Simulationmethoden in der Medizin und Biologie. Medizinische Informatik und Statistik*, 8, 1978.
- S. M. R. Islam, D. Kwak, M. H. Kabir, M. Hossain, et K. Kwak. The internet of things for health care : A comprehensive survey. *IEEE Access*, 3 :678–708, 2015.
- Jannotti. J, DeCouto. D. S. J, Karger. D. R, Morris .R, et Li. J. A scalable location service for geographic ad-hoc routing. Dans *the 6th annual international conference on Mobile computing and networking*, pages 120–130, 2000.
- D. C. Jenn. Transmission equation for multiple cooperative transmitters and collective beamforming. *IEEE Antennas and Wireless Propagation Letters*, 7 :606–608, 2008.
- C. J. jie, X. B. chong, et D. Li. Relative localization systems and algorithms for wireless sensor networks. Dans *2008 IEEE International Conference on Networking, Sensing and Control*, pages 1439–1444, 2008.
- V Kalyani. Enhancing localization accuracy in wireless sensor networks using range-free methods and rss measurements. *2018 International Conference on Recent Trends in Advance Computing (ICRTAC)*, pages 136–142, 2018.
- J. Kang et D. Nyang. A privacy-preserving mobile payment system for mass transit. *IEEE Transactions on Intelligent Transportation Systems*, 18 (8) :2192–2205, 2017.

- A. A. Kannan, Guoqiang Mao, et B. Vucetic. Simulated annealing based wireless sensor network localization with flip ambiguity mitigation. Dans *2006 IEEE 63rd Vehicular Technology Conference*, volume 2, pages 1022–1026, 2006.
- A. Karati, R. Amin, S. K. H. Islam, et K. R. Choo. Provably secure and lightweight identity-based authenticated data sharing protocol for cyber-physical cloud environment. *IEEE Transactions on Cloud Computing*, pages 1–1, 2018a.
- A. Karati, S. H. Islam, et M. Karuppiah. Provably secure and lightweight certificateless signature scheme for iot environments. *IEEE Transactions on Industrial Informatics*, 14(8) :3701–3711, 2018b.
- A. Karimi et S.M. Amini. Reduction of energy consumption in wireless sensor networks based on predictable routes for multi-mobile sink. *J Supercomput*, 75 :7290–7313, 2019.
- Supreet Kaur et Rajiv Mahajan. Hybrid meta-heuristic optimization based energy efficient protocol for wireless sensor networks. *Egyptian Informatics Journal*, 19(3) :145–150, 2018.
- A. Kaveh et A. Dadras. A novel meta-heuristic optimization algorithm : Thermal exchange optimization. *Adv. Eng. Softw.*, 110 :69–84, 2017.
- A. Kaveh et M. Khayatazad. A new meta-heuristic method : Ray optimization. *Computers and Structures*, 112 :283–294, 2012.
- A. Kaveh et S. Talatahari. A novel heuristic optimization method : charged system search. *Acta Mechanica*, 213 :267–289, 2010.
- J. Kennedy et R. Eberhart. Particle swarm optimization. *Proceedings of ICNN'95 - International Conference on Neural Networks*, 4 :1942–1948 vol.4, 1995.
- Soheyl Khalilpourazari et S. Khalilpourazary. Optimization of production time in the multi-pass milling process via a robust grey wolf optimizer. *Neural Computing and Applications*, 29 :1321–1336, 2016.
- Soheyl Khalilpourazari et A. Khamseh. Bi-objective emergency blood supply chain network design in earthquake considering earthquake magnitude : a comprehensive study with real world application. *Annals of Operations Research*, pages 1–39, 2019.
- Ines Khoufi, Pascale Minet, Anis Laouiti, et Saoucene Mahfoudh. Survey of deployment algorithms in wireless sensor networks : Coverage and connectivity issues and challenges. *International journal of autonomous and adaptive communications systems*, 10(4) :341–390, 2017.
- S. Kirkpatrick, C. D. Gelatt, et M. Vecchi. Optimization by simulated annealing. *Science*, 220 :671 – 680, 1983.
- J. Ko, C. Lu, M. B. Srivastava, J. A. Stankovic, A. Terzis, et M. Welsh. Wireless sensor networks for healthcare. *Proceedings of the IEEE*, 98(11) : 1947–1960, 2010.

- Aarti Kochhar et Naresh Kumar. Cwiresless sensor networks for greenhouses : An end-to-end review. *Computers and Electronics in Agriculture*, 163 :104877, 2013.
- J. Koza. Genetic programming - on the programming of computers by means of natural selection. Dans *Complex adaptive systems*, 1993.
- R. V. Kulkarni et G. K. Venayagamoorthy. Bio-inspired algorithms for autonomous deployment and localization of sensor nodes. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 40(6) : 663–675, 2010.
- R. V. Kulkarni, G. K. Venayagamoorthy, et M. X. Cheng. Bio-inspired node localization in wireless sensor networks. Dans *2009 IEEE International Conference on Systems, Man and Cybernetics*, pages 205–210, 2009.
- A. Kumar, A. Khosla, J. S. Saini, et S. Singh. Meta-heuristic range based node localization algorithm for wireless sensor networks. Dans *2012 International Conference on Localization and GNSS*, pages 1–7, 2012.
- Srikanta Kumar. Sensit : Sensor information technology for the warfighter. Dans *Information Fusion* , 2001, pages 3–9, 2001.
- M. Li, H. Zhao, Xingwei Weng, et Tong Han. A novel nature-inspired algorithm for optimization : Virus colony search. *Adv. Eng. Softw.*, 92 : 65–88, 2016a.
- S. Li, Mingkui Tan, I. Tsang, et J. Kwok. A hybrid pso-bfgs strategy for global optimization of multimodal functions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41 :1003–1014, 2011.
- Z. Li et L. Lei. Sensor node deployment in wireless sensor networks based on improved particle swarm optimization. Dans *2009 International Conference on Applied Superconductivity and Electromagnetic Devices*, pages 215–217, 2009.
- Zhiming Li, Yongquan Zhou, Sen Zhang, et Junmin Song. Lévy-flight moth-flame algorithm for function optimization and engineering design problems. *Mathematical Problems in Engineering*, 2016, 2016b.
- Wei Liu, Enqing Dong, et Yang Song. Analysis of flip ambiguity for robust three-dimensional node localization in wireless sensor networks. *Journal of Parallel and Distributed Computing*, 97 :57–68, 2016.
- M. Lones. Metaheuristics in nature-inspired algorithms. *Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation*, 2014.
- Dener M. Wisen : a new sensor node for smart applications with wireless sensor networks. *Comput Electr Eng*, 64 :380–394, 2017.
- Samaneh MahdaviFar et Ali A. Ghorbani. Application of deep learning to cybersecurity : A survey. *Neurocomputing*, 347(C) :149–176, 2019.

- R. Maheswar, A. R. Maria, N. Sheriff, V. Mahima, G. R. Kanagachidambaresan, et M. Lakshmi. Mobility aware next hop selection algorithm (mansa) for wireless body sensor network. Dans *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pages 1–5, 2019.
- M. G. R. Maldonado. Wireless sensor network for smart home services using optimal communications. Dans *2017 International Conference on Information Systems and Computer Science (INCISCOS)*, pages 27–32, 2017.
- G. Mao, B. Fidan, et B.D. Anderson. Wireless sensor network localization techniques. *Computer Networks : The International Journal of Computer and Telecommunications Networking*, 51(10) :2529–2553, 2007a.
- Guoqiang Mao, Barış Fidan, et Brian D.O. Anderson. Wireless sensor network localization techniques. *Computer Networks*, 51(10) :2529 – 2553, 2007b.
- J. Mass-Sanchez, E. Ruiz-Ibarra, J. Cortez-González, A. Espinoza-Ruiz, et Luis A. Castro. Weighted hyperbolic dv-hop positioning node localization algorithm in wsns. *Wireless Pers Commun*, 96 :5011–5033, 2017.
- M. Mihoubi, A. Rahmoun, et P. Lorenz. Metaheuristic rssi based for node localization in distributed wireless sensor network. Dans *2017 Global Information Infrastructure and Networking Symposium (GIIS)*, pages 64–70, 2017.
- Miloud Mihoubi, Abdellatif Rahmoun, et Pascal Lorenz. Moth flame optimization algorithm range-based for node localization challenge in decentralized wireless sensor network. *IJDST*, 10 :82–109, 2019.
- Miloud Mihoubi, Abdellatif Rahmoun, Pascal Lorenz, et Noureddine Lasla. An effective bat algorithm for node localization in distributed wireless sensor network. *Secur. Priv.*, 1 :e20, 2018.
- Miloud Mihoubi, Abdellatif Rahmoun, Meriem Zerkouk, Pascal Lorenz, et Lotfi Baidar. Intelligent technique based on enhanced metaheuristic for optimization problem in internet of things and wireless sensor network. *Int. J. Grid High Perform. Comput.*, 12 :17–42, 2020.
- Pascale Minet, Gerard Chalhoub, Erwan Livolant, Michel Misson, Ridha Soua, Rana Diab, Badr Rmili, et Jean-Francois Perelgritz. Multichannel wireless sensor networks for structural health monitoring of aircraft and launchers. Dans *Wireless sensor systems for extreme environments : space, underwater, underground and industrial*, John Wiley, 2017.
- S. Mirjalili. The ant lion optimizer. *Adv. Eng. Softw.*, 83 :80–98, 2015a.
- S. Mirjalili. Dragonfly algorithm : a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. *Neural Computing and Applications*, 27 :1053–1073, 2015b.
- S. Mirjalili. Sca : A sine cosine algorithm for solving optimization problems. *Knowl. Based Syst.*, 96 :120–133, 2016.

- S. Mirjalili et A. Hatamlou. Multi-verse optimizer : a nature-inspired algorithm for global optimization. *Neural Computing and Applications*, 27 : 495–513, 2015.
- S. Mirjalili et A. Lewis. Grey wolf optimizer. *Adv. Eng. Softw.*, 69 :46–61, 2014.
- S. Mirjalili et A. Lewis. The whale optimization algorithm. *Adv. Eng. Softw.*, 95 :51–67, 2016.
- Seyed Mohammad Mirjalili. Moth-flame optimization algorithm : A novel nature-inspired heuristic paradigm. *Knowl. Based Syst.*, 89 :228–249, 2015c.
- Nissanka B. Priyantha Allen K.L. Miu, Hari Balakrishnan, et Seth Jared Teller. The cricket compass for context-aware mobile applications. Dans *MobiCom '01 : Proceedings of the 7th annual international conference on Mobile computing and networking July 2001*, page 1–14, 2001.
- F. F. Moghaddam, R. Moghaddam, et M. Cheriet. Curved space optimization : A random search based on general relativity theory. *ArXiv*, abs/1208.2214, 2012.
- S. S. Mohar, S. Goyal, et R. Kaur. A survey of localization in wireless sensor network using optimization techniques. Dans *2018 4th International Conference on Computing Communication and Automation (ICCCA)*, pages 1–6, 2018.
- Avinash More et Vijay Raisinghani. A survey on energy efficient coverage protocols in wireless sensor networks. *Journal of King Saud University - Computer and Information Sciences*, 29(4) :428–448, 2017.
- A. Moussa et N. El-Sheimy. Localization of wireless sensor network using bees optimization algorithm. Dans *The 10th IEEE International Symposium on Signal Processing and Information Technology*, pages 478–481, 2010.
- N. Muangkote, K. Sunat, et S. Chiewchanwattana. Multilevel thresholding for satellite image segmentation with moth-flame based optimization. Dans *2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pages 1–6, 2016.
- Dey N, Ashour AS, Shi F, Fong SJ, et Sherratt RS. Developing residential wireless sensor networks for ecg healthcare monitoring. *IEEE Trans Consum Electron*, 63(4) :442–449, 2017.
- Reijers. N et Langendoen. K. Fuzzy weighted centroid localization with virtual node approximation in wireless sensor networks. *Computer Networks*, 43(4) :499–518, 2003.
- D. R. Nair et N. K. Prakash. Design and implementation of a wireless sensor network based efficient led lighting system. Dans *2015 6th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pages 1–6, 2015.

- P. H. Namin et M. A. Tinati. Node localization using particle swarm optimization. Dans *2011 Seventh International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pages 288–293, 2011.
- L. M. Ni. China's national research project on wireless sensor networks. Dans *2008 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (sutc 2008)*, pages 19–19, 2008.
- D. Niculescu et Badri Nath. Ad hoc positioning system (aps) using aoa. Dans *IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No.03CH37428)*, volume 3, pages 1734–1743 vol.3, 2003.
- Il-Seok Oh, Jin-Seon Lee, et B. Moon. Hybrid genetic algorithms for feature selection. *IEEE transactions on pattern analysis and machine intelligence*, 26 11 :1424–37, 2004.
- B.G Paik, S.R Cho, B.J Park, D. Lee, B.D Bae, et J.H Yun. Characteristics of wireless sensor network for full scale ship application. *Journal of Marine Science and Technology*, 14(1) :115–126, 2009.
- Sanjaya Kumar Panda, Sohan Kumar Pande, et Satyabrata Das. Task partitioning scheduling algorithms for heterogeneous multi-cloud environment. *Arabian Journal for Science and Engineering*, 43 :913–933, 2018.
- S.K. Panda et P.K. Jana. Normalization-based task scheduling algorithms for heterogeneous multi-cloud environment. *Inf Syst Front*, 20(2) : 373–399, 2018.
- S.K. Panda et P.K. Jana. An energy-efficient task scheduling algorithm for heterogeneous cloud computing systems. *Cluster Comput*, 22(2) : 509–527, 2019.
- M. M. Patil, U. Shaha, U. B. Desai, et S. N. Merchant. Localization in wireless sensor networks using three masters. Dans *2005 IEEE International Conference on Personal Wireless Communications, 2005. ICPWC 2005.*, pages 384–388, 2005.
- N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses, et N. S. Correal. Locating the nodes : cooperative localization in wireless sensor networks. *IEEE Signal Processing Magazine*, 22(4) :54–69, 2005.
- A.K. Paul et T. Sato. Localization in wireless sensor networks : A survey on algorithms, measurement techniques, applications and challenges. *J. Sens. Actuator Netw*, 6(24), 2017.
- Ayona Philipose et A Thakker Rajesh. Investigation on energy efficient sensor node placement in railway systems. *Engineering Science and Technology, an International Journal*, 19 :754–768, 2016.
- S. Phoemphon, C. So-In, et N. Leelathakul. Fuzzy weighted centroid localization with virtual node approximation in wireless sensor networks. *IEEE Internet of Things Journal*, 5(6) :4728–4752, 2018.

- R. Poli, J. Kennedy, et T. Blackwell. Particle swarm optimization. *Swarm Intelligence*, 1 :33–57, 2007.
- N. B. Priyantha, A. Chakraborty, et H. Balakrishnan. The cricket location-support system. Dans *MobiCom '00 : Proceedings of the 6th annual international conference on Mobile computing and networking August 2000*, page 32–43, 2000.
- [N. B. Priyantha, Allen K. L. Miu, H. Balakrishnan, et S. Teller. The cricket compass for context-aware mobile applications. Dans *MobiCom '01 : Proceedings of the 7th annual international conference on Mobile computing and networking July 2001*, page 1–14, 2001.
- S. Pundir, M. Wazid, D. P. Singh, A. K. Das, J. J. P. C. Rodrigues, et Y. Park. Intrusion detection protocols in wireless sensor networks integrated to internet of things deployment : Survey and future challenges. *IEEE Access*, 8 :3343–3363, 2020.
- Jun Qi, Po Yang, Geyong Min, Oliver Amft, Feng Dong, et Lida Xu. Privacy preservation in wsn for healthcare application. *Pervasive and Mobile Computing*, 41(C) :132–149, 2017.
- Xiaozhi Qu. *Detection and update of landmarks with uncertainty analysis*. PhD thesis, Université Paris-Est, 2017.
- N. Rai et R. Daruwala. Node density optimisation using composite probabilistic sensing model in wireless sensor networks. *IET Wireless Sensor Systems*, 9(4) :181–190, 2019.
- R. Rajakumar, J. Amudhavel, P. Dhavachelvan, , et T. Vengattaraman. Gwo-lpwsn : Grey wolf optimization algorithm for nodelocalization problem in wireless sensor networks. *Journal of Computer Networks and Communications*, 2017 :7348141, 2017.
- Anagha Rajput et Vinoth Babu Kumaravelu. Scalable and sustainable wireless sensor networks for agricultural application of internet of things using fuzzy c-means algorithm. *Sustainable Computing : Informatics and Systems*, 22 :62–74, 2019.
- E. Rashedi, H. Nezamabadi-pour, et S. Saryazdi. Gsa : A gravitational search algorithm. *Inf. Sci.*, 179 :2232–2248, 2009.
- D. D. K. Rathinam, D. Surendran, A. Shilpa, A. S. Grace, et J. Sherin. Modern agriculture using wireless sensor network (wsn). Dans *2019 5th International Conference on Advanced Computing Communication Systems (ICACCS)*, pages 515–519, 2019.
- F. Reichenbach, J. Salzmann, D. Timmermann, A. Born, et R. Bill. Dls : A resource-aware localization algorithm with high precision in large wireless sensor networks. Dans *2007 4th Workshop on Positioning, Navigation and Communication*, pages 247–254, 2007.
- M. H. Riaz, U. Rashid, M. Ali, et L. Li. Internet of things based wireless patient body area monitoring network. Dans *2017 IEEE International*

- Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pages 970–973, 2017.
- D. Rodenas-Herraiz, A. Garcia-Sanchez J., F. Garcia-Sanchez, et J. Garcia-Haro. Current trends in wireless mesh sensor networks : a review of competing approaches. *Sensors (Basel, Switzerland)*, 13(5) :5958–5995, 2013.
- D. Rojas et J. Barrett. A hardware-software wsn platform for machine and structural monitoring. Dans *2017 28th Irish Signals and Systems Conference (ISSC)*, pages 1–6, 2017.
- Satyabrata Roy, Jyotirmoy Karjee, U.S. Rawat, Dayama Pratik N., et Nilanjan Dey. Symmetric key encryption technique : A cellular automata based approach in wireless sensor networks. *Procedia Computer Science*, 78 :408–414, 2016.
- Selmic R.R., Phoha V.V., et Serwadda A. Wsn platforms. Dans *Wireless Sensor Networks Springer, Cham*, 2016.
- Fong S, Li J, Song W, Tian Y, Wong RK, et Dey N. Predicting unusual energy consumption events from smart home sensor network by data stream mining with misclassified recall. *J Ambient Intell Humaniz Comput*, 9(4) :1197–1221, 2018.
- M. A. Sahi, H. Abbas, K. Saleem, X. Yang, A. Derhab, M. A. Orgun, W. Iqbal, I. Rashid, et A. Yaseen. Privacy preservation in e-healthcare environments : State of the art and future directions. *IEEE Access*, 6 :464–478, 2018.
- Hamid Salimi. Stochastic fractal search : A powerful metaheuristic algorithm. *Knowl. Based Syst.*, 75 :1–18, 2015.
- C. Savarese, J. M. Rabaey, , et K. Langendoen. Robust positioning algorithms for distributed ad-hoc wireless sensor networks. Dans *ATEC '02 : Proceedings of the General Track of the annual conference on USENIX Annual Technical Conference*, page 317–327, 2002.
- A. Savvides et C.-C. Han andand M. B. Strivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. Dans *7th ACM/IEEE Int'l. Conf. Mobile Computing and Networking, Rome, Italy , 2001*, page 166–179, 2001.
- Yoshikazu Sawaragi, Hirotaka Nakayama, et T.T. Theory of multiobjective optimization. Dans *Academia Press, Inc. : London, UK*, volume 176, 1985.
- Fouzi Semchedine, Louiza Bouallouche-Medjkoune, Leila Bennacer, Naim Aber, et Djamil Aïssani. Routing protocol based on tabu search for wireless sensor networks. *Wireless Pers Commun*, 67 :105–112, 2011.
- E. Q. Shahra, T. R. Sheltami, et E. M. Shakshuki. A comparative study of range-free and range-based localization protocols for wireless sensor network : Using cooja simulator. *International Journal of Distributed Systems and Technologies (IJ DST)*. IGI Global, 8(1) :1–16, 2017.

- Nidhi Sharma et Ravindara Bhatt. Privacy preservation in wsn for health-care application. *Procedia Computer Science*, 132 :1243–1252, 2018.
- V. Sharma et D. S. Saini. Performance investigation of advanced multi-hop and single-hop energy efficient leach protocol with heterogeneous nodes in wireless sensor networks. Dans *2015 Second International Conference on Advances in Computing and Communication Engineering*, pages 192–197, 2015.
- D. Simon. Biogeography-based optimization. *IEEE Transactions on Evolutionary Computation*, 12 :702–713, 2008.
- D. Simon. Biogeography-based optimization. *IEEE Transactions on Evolutionary Computation*, 12(6) :702–713, 2008.
- A. Simonetto et G. Leus. Distributed maximum likelihood sensor network localization. *IEEE Transactions on Signal Processing*, 62(6) :1424–1437, 2014.
- A. K. Singh et P. Kumar. Advancement in quality of services in wireless sensor networks. Dans *2018 3rd International Conference On Internet of Things : Smart Innovation and Usages (IoT-SIU)*, pages 1–5, 2018.
- S. Singh, Shivangna, et E. Mittal. Range based wireless sensor node localization using pso and bbo and its variants. Dans *2013 International Conference on Communication Systems and Network Technologies*, pages 309–315, 2013.
- Konstantinos Skiadopoulos, Athanasios Tsipis, Konstantinos Giannakis, George Koufoudakis, Eleni Christopoulou, Konstantinos Oikonomou, George Kormentzas, et Ioannis Stavrakakis. Synchronization of data measurements in wireless sensor networks for iot applications. *Ad Hoc Networks*, 89 :47–57, 2019.
- Abdallah Sobehy, Eric Renault, et Paul Muhlethaler. Position certainty propagation : a location service for manets. Dans *MSPN 2018 - 4th International Conference on Mobile, Secure and Programmable Networking, Jun 2018, Paris, France*, pages 131–142, 2018.
- K. Steinhöfel, A. Albrecht, et C. Wong. Convergence analysis of simulated annealing-based algorithms solving flow shop scheduling problems. Dans *CIAC*, 2000.
- R. Storn et K. Price. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11 :341–359, 1997.
- K. Taha. Methods that optimize multi-objective problems : A survey and experimental evaluation. *IEEE Access*, 8 :80855–80878, 2020.
- S. B. Takale et S. D. Lokhande. Quality of service requirement in wireless sensor networks : A survey. Dans *2018 IEEE Global Conference on Wireless Computing and Networking (GCWCN)*, pages 34–38, 2018.

- AEI-Ghazali Talbi. Machine learning into metaheuristics : A survey and taxonomy of data-driven metaheuristics. *ACM Comput. Surv.*, 00 :30, 2020.
- E. Talbi. Metaheuristics - from design to implementation. Dans *Wiley*, 2009.
- Rong Tan, Yudong Li, Yifan Shao, et Wen Si. Distance mapping algorithm for sensor node localization in wsns. *International Journal of Wireless Information Networks*, 27 :261–270, 2020.
- J. Tillett, R. Rao, et F. Sahin. Cluster-head identification in ad hoc sensor networks using particle swarm optimization. Dans *2002 IEEE International Conference on Personal Wireless Communications*, pages 201–205, 2002.
- T. Ting, K. Wong, et C. Chung. Investigation of hybrid genetic algorithm/-particle swarm optimization approach for the power flow problem. *2005 International Conference on Machine Learning and Cybernetics*, 1 :436–440, 2005a.
- T. Ting, K. P. Wong, et C. Chung. A hybrid genetic algorithm/particle swarm approach for evaluation of power flow in electric network. Dans *ICMLC*, 2005b.
- T. O. Ting, Xin-She Yang, Shi Cheng, et Kaizhu Huang. Hybrid metaheuristic algorithms : Past, present, and future. Dans *Recent Advances in Swarm Intelligence and Evolutionary Computation*, 2015.
- A. O. Topal et Oguz Altun. A novel meta-heuristic algorithm : Dynamic virtual bats algorithm. *Inf. Sci.*, 354 :222–235, 2016.
- K. Trivedi et A. K. Srivastava. An energy efficient framework for detection and monitoring of forest fire using mobile agent in wireless sensor networks. Dans *2014 IEEE International Conference on Computational Intelligence and Computing Research*, pages 1–4, 2014.
- J. Tsai, T. Liu, et J. Chou. Hybrid taguchi-genetic algorithm for global numerical optimization. *IEEE Transactions on Evolutionary Computation*, 8 :365–377, 2004.
- M. J. Varnamkhashti. A hybrid of adaptive neuro-fuzzy inference system and genetic algorithm. *J. Intell. Fuzzy Syst.*, 25 :793–796, 2013.
- Massimo Vecchio, Roberto López-Valcarce, et Francesco Marcelloni. A two-objective evolutionary approach based on topological constraints for node localization in wireless sensor networks. *Appl. Soft Comput.*, 12 :1891–1901, 2012.
- T. Voigt, L. Mottola, et K. Hewage. Understanding link dynamics in wireless sensor networks with dynamically steerable directional antennas. Dans : *Demeester P., Moerman I., Terzis A. (eds) Wireless Sensor Networks. EWSN 2013. Lecture Notes in Computer Science*, volume 7772, pages 115–130, 2013.

- Y. Wang, Jinqun Hang, C. Li, J. You, Shujia Chen, et L. Cheng. A mobile localization method for wireless sensor network in mixed los/nlos environments. *2018 8th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, pages 87–90, 2018.
- A. Ward, A. Jones, et A. Hopper. A new location technique for the active office. *IEEE Personal Communications*, 4(5) :42–47, 1997.
- Wei Ye, J. Heidemann, et D. Estrin. An energy-efficient mac protocol for wireless sensor networks. Dans *Proceedings.Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 3, pages 1567–1576 vol.3, 2002.
- J. Widmer, Harttenstein. H, et Mauve. M. A survey on position based routing in mobile ad-hoc networks. *Network Magazine*, 15 :30–39, 2001.
- D. Wolpert et W. Macready. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.*, 1 :67–82, 1997.
- D. H. Wolpert et W. G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1) :67–82, 1997.
- K. . S. Wong, I. W. Tsang, V. Cheung, S. . G. Chan, et J. T. Kwok. Position estimation for wireless sensor networks. Dans *GLOBECOM '05. IEEE Global Telecommunications Conference, 2005.*, volume 5, pages 5 pp.–2776, 2005.
- Li X, Keegan B, et Mtenzi F. Energy efficient hybrid routing protocol based on the artificial fish swarm algorithm and ant colony optimisation for wsns. *Sensors (Basel)*, 18(10) :3351, 2018.
- Xiao-Feng Xie, W. Zhang, et Zhilian Yang. Dissipative particle swarm optimization. *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)*, 2 :1456–1461 vol.2, 2002.
- X. Yang et A. Gandomi. Bat algorithm : a novel approach for global engineering optimization. *Engineering Computations*, 29 :464–483, 2012.
- X. Yang, M. Karamanoglu, T. Ting, et Y. Zhao. Applications and analysis of bio-inspired eagle strategy for engineering optimization. *Neural Computing and Applications*, 25 :411–420, 2013a.
- X. Yang, T. Ting, et M. Karamanoglu. Random walks, lévy flights, markov chains and metaheuristic optimization. Dans *John Wiley and Sons, Inc.*, 2013b.
- Xin-She Yang et S. Deb. Cuckoo search via lévy flights. *2009 World Congress on Nature and Biologically Inspired Computing (NaBIC)*, pages 210–214, 2009.
- L. Yong-Min, W. Shu-Ci, et N. Xiao-Hong. The architecture and characteristics of wireless sensor network. Dans *2009 International Conference on Computer Technology and Development*, volume 1, pages 561–565, 2009.

- X. Yuan, P. He, Q. Zhu, et X. Li. Adversarial examples : Attacks and defenses for deep learning. *IEEE Transactions on Neural Networks and Learning Systems*, 30(9) :2805–2824, 2019.
- Yunfeng Han, Cuie Zheng, et Dajun Sun. Localization of large scale underwater sensor networks based on recursive position estimation. Dans *OCEANS 2015 - MTS/IEEE Washington*, pages 1–4, 2015.
- M. M. Zanjireh et H. Larijani. A survey on centralised and distributed clustering routing algorithms for wsns. Dans *2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*, pages 1–6, 2015.
- G. Zhang et Y. Wang. Optimizing coordinated ramp metering : A preemptive hierarchical control approach. *Comput. Aided Civ. Infrastructure Eng.*, 28 :22–37, 2013.
- W. Zhang et Xiao-Feng Xie. Depso : hybrid particle swarm with differential evolution operator. *SMC'03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference Theme - System Security and Assurance (Cat. No.03CH37483)*, 4 :3816–3821 vol.4, 2003.
- Jungang Zheng, Chengdong Wu, Hao Chu, et Yang Xu. An improved rssi measurement in wireless sensor networks. *Procedia Engineering*, 15 : 876–880, 2011.
- Pang. Zhibo. *Technologies and Architectures of the Internet-of-Things (IoT) for Health and Well-being*. PhD thesis, Institut royal de technologie KTH : Kista Suède,, 2013.
- J. Zhu, X. Liu, Q. Shi, T. He, Z. Sun, X. Guo, W. Liu, O. B. Sulaiman, B. Dong, et C. Lee. Development trends and perspectives of future sensors and mems/nems. *Micromachines*, 11(1) :7, 2019.

### إرشادية تعريفية هجينة لمشكلة التحسين في شبكات الاستشعار اللاسلكية

لقد عرفت دراسة شبكات الاستشعار اللاسلكية مؤخراً اهتماماً منقطع النظير و هذا راجع الى الحاجة الماسة لتطبيقاتها العديدة في الميادين ذات الانتشار المكثف وعلى مساحة كبيرة. يعتبر تحديد موقع العقد واحدة من المشاكل المهمة في شبكات الاستشعار اللاسلكية، والقدرة على تحديد موقع العقد مرغوب فيه للغاية لتقييم الأداء في تطبيقات المراقبة. يُعرّف الموقع بأنه تقدير لمواقع أجهزة الاستشعار التي تحتوي على معلومات، تتمحور العملية بالاستعانة بأجهزة معلومة لتحديد موقع البقية لأن معظم أجهزة الاستشعار لا تعرف موقعها بسبب تكلفة أجهزة الاستشعار المكلفة وحجمها. الهدف الرئيسي للموقع هو تحديد إحداثيات العقد المجهولة في أقل وقت و بأقل تكاليف، لهذا السبب و من أجله، تُقترح الأساليب الحديثة القائمة على تقنيات الذكاء الاصطناعي حيث يعتبر موقع العقدة مشكلة تحسين في الفضاء كمشكل متعدد الأبعاد. في الآونة الأخيرة، تم اقتراح خوارزمية لهب العثة. تقترح هذه الورقة خوارزمية لحل مشكلة توطين العقد، قوم الخوارزمية بحساب (حسب التطور) مواقع الحفافيش بناء على العقد متكررة عبر المسافة الإقليدية. نشر هذه الخوارزمية على شبكات الاستشعار اللاسلكية ذات نطاق واسع مع مئات أجهزة الاستشعار يدل على فعالية و كفاءة الأداء اللائق لتحديد موقع العقدة. بالإضافة إلى ذلك، يتم محاكاة و دراسة عوامل و مؤشرات الخوارزمية مع تفسير للتأثيرات المحصلة في سيناريوهات محاكاة مختلفة، زد على ذلك فقد أجريت دراسة مقارنة لإثبات أداء الخوارزمية المقترحة مع الخوارزميات الحديثة و ذات الصيت الذائع، و قد أظهرت نتائج المحاكاة أن الخوارزمية المقترحة لديها معدل تقارب ودقة أكبر مقارنة بخوارزميات الأخرى مثل خوارزمية تحسين سرب الجسيمات، خوارزمية التطور التفاضلي، خوارزمية تلقيح الزهرة، خوارزمية الحفافيش، ... إلخ.

#### Résumé

Les réseaux de Capteurs sans Fils (RCSFs) ont récemment fait l'objet d'études approfondies en raison de leurs nombreuses applications dans des processus qui doivent être répartis sur une grande surface. L'un des problèmes importants du RCSF est la localisation des nœuds, la capacité de localisation des nœuds est hautement souhaitable pour l'évaluation des performances dans les applications de surveillance. La localisation est définie comme l'estimation des emplacements de capteurs avec des informations de localisation initialement inconnues, car la plupart des capteurs ne connaissent pas leur emplacement en raison du coût et de la taille des capteurs. L'objectif principal de la localisation est de déterminer les coordonnées des nœuds inconnus dans un temps minimal et avec un faible coût énergétique, pour cette raison, des approches récentes basées sur des techniques d'intelligence sont proposées et la localisation des nœuds est considérée comme un problème d'optimisation dans un espace multidimensionnel. Récemment, l'algorithme de méta-heuristique de la flamme des

mites a été proposé comme solution pour les problèmes d'optimisation. Ce document propose un algorithme d'optimisation de la flamme des mites pour résoudre le problème de localisation des nœuds, l'algorithme d'optimisation de la flamme des mites calcule (par l'évolution) les positions des nœuds itérativement à travers la distance euclidienne comme forme physique. Déployer cet algorithme dans un réseau avec des centaines de capteurs représente un challenge qui prouve la performance décente en matière de convergence et en matière de ressources consommées pour la localisation des nœuds. De plus, les paramètres de l'algorithme d'optimisation de la flamme des mites sont simulés et interprétés dans différents scénarios de simulation, en outre, une étude comparative a été réalisée pour démontrer les performances de l'algorithme proposé, les résultats de simulation obtenus prouvent que l'algorithme a une vitesse de convergence et avec plus précision par rapport aux algorithmes de : chauve-souris, optimisation de l'essaim de particules et Évolution différentielle, etc.

### Abstract

Wireless Sensor Networks (RCSFs) have recently been extensively studied because of their many applications in processes that need to be distributed over a large area. One of the important problems of RCSF is the location of the nodes, the ability to locate nodes is highly desirable for evaluating performance in monitoring applications. Location is defined as the estimation of sensor locations with initially unknown location information since most sensors do not know their location due to the cost and size of the sensors. The main objective of the location is to determine the coordinates of the unknown nodes by a minimal time and with a low energy cost, for this reason, recent approaches based on intelligence techniques are proposed and node location is considered an optimization problem in a multidimensional space. Recently, the Moth Flame meta-heuristic algorithm has been proposed as a solution for the optimization problem. This paper proposes a Moth Flame algorithm to solve the problem of node localization, Moth Flame algorithm calculates (by the evolution) the positions of nodes in iterative manner across the euclidean distance as a physical form. Deploying this algorithm in a network with hundreds of sensors represents a challenge that proves the decent performance in terms of convergence and in terms of resources consumed for locating nodes. In addition, the Moth Flame parameters are simulated and interpreted in different simulation scenarios, in addition, a comparative study was carried out to demonstrate the performance of the proposed algorithm and the simulation results obtained prove that the algorithm has a good convergence and with more precision compared to the algorithms: Bat ( Bat algorithm ), PSO (particle swarm optimization) and DE (Differential Evolution), etc.