# République Algérienne Démocratique et Populaire Ministère de l'Enseignement Supérieure et de la Recherche Scientifique Université Ahmed Draia - Adrar Faculté des Sciences et de la Technologie Département des Mathématiques et Informatique



Mémoire de fin d'étude, en vue de l'obtention du diplôme de Master En informatique

**Option: Système Intelligents** 

Thème

# Une méta-heuristique hybride pour une meilleure localisation des nœuds dans les RCSFs.

Préparés par

BENSMAIL MOULAY ALI CHERIF LILA MOATEZ

#### Devant le jury :

Encadreur : Dr. RABHI SEDDIK
Président : Dr.KADDI MOHAMMED
Examinateur : Pr.OMARI MOHAMMED

Année Universitaire 2020/2021

#### **Remerciements**

Nous remercions tout d'abord le bon Dieu, le tout puissant de nous avoir armé de force et de courage pour mener à terme ce projet.

Merci à tous ceux qui ont contribué à ce que la réalisation de ce projet soit possible.

Notre profonde gratitude s'adresse à Monsieur **RABHI SEDDIK**, encadreur

de ce mémoire pour ses conseils fructueux et pour son aide précieuse qui nous a conduits à concrétiser ce travail.

Notre profonde gratitude s'adresse à Monsieur **KOHILI** et Monsieur **OMARI** et Monsieur **KADDI** pour leur grande aide dans ce travail et pour tous ce qu'ils ont fait pour nous.

Enfin, nous ne pouvons pas s'empêcher de remercier tout le corps enseignant de département Mathématique et Informatique de l'université Ahmed Draya ADRAR pour la qualité d'enseignement qu'il nous a offert et d'avoir bâti l'édifice intellectuel que nous sommes d'ores et déjà.

### **Dédicace**

Je dédie ce modeste travail

• mon père et ma mère qui se sont sacrifiés pour que je puisse achever mes études.

Ainsi que mes oncles, mes tantes et mes grands parents.

• toute la famille.

• tous mes amies, à tous les enseignants qui m'ont aidé de prés ou de loin à obtenir mon licence en informatique.

#### Résumé

Les travaux présentés dans cette thèse s'inscrivent dans le cadre général des réseaux de capteurs sans fil. Ils se concentrent sur l'étude et la mise en œuvre de nouveaux algorithmes distribués qui utilisent des heuristiques inspirées de la biologie pour résoudre le problème de localisation. Dans cette mémoire, nous avons proposé une m l'algorithme d'optimisation par fruit fly FOA avec l'algorithme d'optimisation d'inférence pour optimiser le sitemap estimé par une technique de localisation célèbre (Centroid) afin de fournir la localisation de deux nouvelles approches hybrides. Pour valider les performances de nos approches proposées, nous avons mené des expériences et les avons comparées avec l'algorithme optimisé () pour évaluer le taux d'amélioration.

**Mots clés :** réseaux de capteurs sans fil, localisation, métadonnées, optimisation, mouche des fruits.

#### لخص

يقع العمل المقدم في هذه الرسالة ضمن الإطار العام الشبكات الاستشعار اللاسلكية. يركزون على دراسة وتنفيذ الخوار زميات الموزعة الجديدة التي تستخدم الاستدلال البيولوجي المستوحى من علم الأحياء لحل مشكلة التوطين. في هذه الرسالة، قمنا بتكييف الاستدلال الفوقي لخوار زمية تحسين ذبابة الفاكهة (FOA) مع خوار زمية تحسين الاستدلال التحسين خريطة المواقع المقدرة بتقنية توطين شهيرة (Centroid) من أجل توفير توطين هجينين جديدين اقتراب. المتحقق من صحة أداء مناهجنا المقترحة، أجرينا تجارب وقارناها بخوار زمية (CSO) المحسنة لتقييم معدل التحسين.

الكلمات المفتاحية: شبكات الاستشعار اللاسلكية، التعريب، الفوقية، التحسين، ذبابة الفاكهة.

#### summary

The work presented during this thesis falls within the overall framework of wireless sensor networks. They specialise in the study and implementation of latest distributed algorithms that use biologically inspired heuristics to unravel a localization problem. During This thesis, we've adapted the inference optimization algorithm of Drosophila FOA with the inference optimization algorithm to optimize the sitemap estimated by a famous localization technique (Centroid) so as to supply the localization of two new hybrideapproaches. To validate the performance of our proposed approaches, we conducted experiments and compared them with the optimized algorithm () to assess the improvement rate.

**Keywords**: wireless sensor networks, location, metadata, optimization, fruit fly.

#### **Sommair**

Remerci	ements	2
Dédicace	· · · · · · · · · · · · · · · · · · ·	3
Résumé		4
Sommai	r	5
Liste des	figures	8
Liste des	Tableaux	9
Introduc	tion générale	. 10
СНАРІТ	TRE 1	. 12
1. Défi	nition :	. 13
2. Les	éléments constitutifs d'un RCSF:	. 13
3. Qu'	est-ce qu'un capteur :	. 14
3.1.	Définition :	. 14
3.2.	Architecteur et composition d'un capteur sans fil :	. 14
•	L'unité d'acquisition	. 14
•	L'unité de traitement	. 15
•	L'unité de communication	. 15
•	Batterie	. 15
•	La mobilité	. 15
3.3.	Exemples des types de capteurs :	. 15
4. Les	réseaux de capteurs sans fil	. 17
4.1.	Définition d'un RCSF	. 17
4.2.	Le déploiement des nœuds du RCSF	. 17
4.3.	Caractéristique des RCSF :	. 17
4.4.	Cas d'application des RCSF :	. 17
4.4.	1. Feux de forêt :	. 18
4.4.	2. Domaine militaire :	. 18
.4.5	Les limites des RCSF :	. 18
Conclusi	on	. 19
СНАРІТ	TRE 2	. 20
1. Intr	oduction	. 21
2. Défi	nition d'un système de localisation :	. 21
3. Syst	ème de localisation :	. 21
3.1	Critères d'un système de localisation :	. 21
3.1.	l Précision de la localisation :	. 21

	3.1.	2 Contraintes de ressources :	21
	3.1.	3 Contraintes énergétiques :	21
	3.1.4	Passage à l'échelle :	21
4.	La l	ocalisation dans les RCSF	22
	4.1	Estimation des distances	22
	4.2	Dérivation des positions	22
5.	Tec	hniques de mesure :	23
	5.1.	Mécanismes de dérivation des positions :	23
	5.1.	1. La trilatération :	23
	5.1.	2. La triangulation	24
	5.1.	3. Multilatération :	24
	<b>5.2.</b>	Modèle d'estimation des distances :	26
	<b>❖</b> B	ilan des techniques de mesures	26
	5.3.	Un aperçu des algorithmes de localisation existants	26
	5.3.	1. DV-HOP et DV-distance :	26
	5.3.	2. APIT:	27
	5.3.	3. Centroid:	27
C	onclusi	on	28
C]	HAPIT	TRE 3	29
1.	Intr	oduction:	30
2.	Pro	blème d'optimisation :	30
3.	Mét	a-heuristique d'optimisation :	30
4.	Opt	imisation par PSO :	30
5.	Opt	imisation par Fruit Fly :	31
	5.1.	Aspect biologique :	31
C	onclusi	on :	34
C]	HAPIT	TRE 4	35
1.	Intr	oduction :	36
2.	Le l	angage de programmation utilisé :	36
3.	Cen	troid :	37
4.	For	mulation de problème :	38
5.	Des	cription générale de l'approche de localisation proposé	38
6.	Les	étapes de la simulation :	41
7.	Des	cription des étapes d'exécution d'application :	41
	7.1.	Choix des paramètres de réseau :	42
	7.2.	Après de saisir les paramètres de réseau :	42
	7.3.	Choix du paramètre de FOA et PSO :	42

<b>7.4.</b>	Lo	caliser par l'algorithm proposé :	42
7.5.	L'	Affichage des informations supplémentaire :	43
8. Év	aluat	tion de l'algorithme proposé	43
8.1.	La	précision de l'algorithme	43
8.2.	Év	aluation de notre application	43
8.2	2.1.	La précision par rapport le nombre des capteurs :	44
8.2	2.2.	Précision par rapport au rayon de connectivite	44
8.2	2.3.	La précision par rapport aux nombre des ancres	45
8.2	2.4.	La précision par rapport la population :	45
8.2	2.5.	La précision par rapport le nombre d'itération	46
Conclu	sion	·	48
Conclu	sion ş	générale	49
Bibliog	raph	ie	50

#### **LISTE DES FIGURES**

CH.	A D	מיויו	1 7 1
١.п.	AP	IIK	

FIGURE 2: EXEMPLE DE CAPTEUR SANS FIL         14           FIGURE 3: ARCHITECTURE MATERIELLE D'UN CAPTEUR SANS FIL         14           FIGURE 4: SCHEMA REPRESENTANT QUELQUES TYPES DE CAPTEURS         16           FIGURE 5: EVOLUTION DES CAPTEURS         16           CHAPITRE 2           CHAPITRE 2           FIGURE2- 1: PROCESSUS DE LOCALISATION         22           FIGURE2- 2: ESTIMATION DE POSITION PAR TRILATERATION         22           FIGURE2- 3: ESTIMATION DE POSITION PAR TRIANGULATION         22           FIGURE2- 4: PRINCIPE DE LA METHODE APIT         2*           CHAPITRE 3           FIGURE 3- 1STRUCTURE CORPORELLE DE LA DROSOPHILE         3*           FIGURE 3- 2SCHEMA DE RECHERCHE DE LA DROSOPHILE         3*           FIGURE 3- 3 ORGANIGRAMME DE L'ALGORITHME FOA         3*           CHAPITRE 4           CHAPITRE 4           CHAPITRE 4           FIGURE 4- 1 L'INTERFACE DU MATLAB         3*           FIGURE 4- 3 L'ORGANIGRAMME PROPOSE         4*           FIGURE 4- 3 L'ORGANIGRAMME PROPOSE         4*           FIGURE 4- 4 LES ETAPES DE SIMULATION         4*           FIGURE 4- 5 L'INTERFACE DE SIMULAT		
FIGURE 3: ARCHITECTURE MATERIELLE D'UN CAPTEUR SANS FIL.         14           FIGURE 4: SCHEMA REPRESENTANT QUELQUES TYPES DE CAPTEURS.         16           FIGURE 5: EVOLUTION DES CAPTEURS.         16           CHAPITRE 2           CHAPITRE 2           FIGURE2- 1: PROCESSUS DE LOCALISATION           2: ESTIMATION DE POSITION PAR TRILATERATION.           2: FIGURE2- 3: ESTIMATION DE POSITION PAR TRIANGULATION.           2: CHAPITRE 3           CHAPITRE 3           FIGURE 3- 1STRUCTURE CORPORELLE DE LA DROSOPHILE           3: FIGURE 3- 2SCHEMA DE RECHERCHE DE LA DROSOPHILE           3: FIGURE 3- 3 ORGANIGRAMME DE L'ALGORITHME FOA           CHAPITRE 4           CHAPITRE 4           FIGURE 4-1 L'INTERFACE DU MATLAB.           3: FIGURE 4-2 LA COORDONE ESTIME DE NOEUD INCONNU PAR CENTROID           3: FIGURE 4-3 L'ORGANIGRAMME PROPOSE           4-4           FIGURE 4-5 L'INTERFACE DE SIMULATION           4-5 FIGURE 4-7 ERREURE DE LOCALISATION PAR RAPPORT NOMBRE DE CAPTEURS           4-7 ERREURE DE LOCALISATION PAR RAPPORT NOMBRE DE CAPTEURS           4-7 ERREURE DE LOCALISATION PAR RAPPORT NOMBRE DE CAPTEURS           4-7 ERREURE DE	FIGURE 1: ARCHITECTURE D'UN RCSF	
FIGURE 4: SCHEMA REPRESENTANT QUELQUES TYPES DE CAPTEURS         16           FIGURE 5: EVOLUTION DES CAPTEURS         16           CHAPITRE 2           FIGURE2- 1: PROCESSUS DE LOCALISATION         22           FIGURE2- 2: ESTIMATION DE POSITION PAR TRIANGULATION         22           FIGURE2- 4: PRINCIPE DE LA METHODE APIT         2*           CHAPITRE 3           FIGURE 3- ISTRUCTURE CORPORELLE DE LA DROSOPHILE         3           FIGURE 3- 2SCHEMA DE RECHERCHE DE LA DROSOPHILE         3           FIGURE 3- 3 ORGANIGRAMME DE L'ALGORITHME FOA         3           CHAPITRE 4           CHAPITRE 4           CHAPITRE 4           FIGURE 4-1 L'INTERFACE DU MATLAB         3*           FIGURE 4-2 LA COORDONE ESTIME DE NOEUD INCONNU PAR CENTROID         3*           FIGURE 4-3 L'ORGANIGRAMME PROPOSE         4           FIGURE 4-4 LES ETAPES DE SIMULATION         4           FIGURE 4-5 L'INTERFACE DE SIMULATION         4           FIGURE 4-5 L'ALCOALISATION AVEC L'ALGORITHM PROPOSE         4           FIGURE 4-7 ERREURE DE LOCALISATION PAR RAPPORT NOMBRE DE CAPIEURS         4           FIGURE 4-7 ERREURE DE L		
CHAPITRE 2		
### CHAPITRE 2    FIGURE 2- 1: PROCESSUS DE LOCALISATION		
FIGURE 2- 1: PROCESSUS DE LOCALISATION         22           FIGURE 2- 2: ESTIMATION DE POSITION PAR TRILATERATION         22           FIGURE 2- 3: ESTIMATION DE POSITION PAR TRIANGULATION         24           FIGURE 2- 4: PRINCIPE DE LA METHODE APIT         2           CHAPITRE 3           FIGURE 3- 1STRUCTURE CORPORELLE DE LA DROSOPHILE         3           FIGURE 3- 2SCHEMA DE RECHERCHE DE LA DROSOPHILE         3           FIGURE 3- 3 ORGANIGRAMME DE L'ALGORITHME FOA         3           CHAPITRE 4           CHAPITRE 4           FIGURE 4- 1 L'INTERFACE DU MATLAB         3           FIGURE 4- 2 LA COORDONE ESTIME DE NOEUD INCONNU PAR CENTROID         3           FIGURE 4- 3 L'ORGANIGRAMME PROPOSE         44           FIGURE 4- 5 L'INTERFACE DE SIMULATION         4           FIGURE 4- 6 LA LOCALISATION AVEC L'ALGORITHM PROPOSE         44           FIGURE 4- 6 LA LOCALISATION AVEC L'ALGORITHM PROPOSE         44           FIGURE 4- 7 ERREURE DE LOCALISATION PAR RAPPORT NOMBRE DE CAPITEURS         44           FIGURE 4- 8 ERREURE DE LOCALISATION PAR RAPPORT RAYON DE CONNECTIVITE         44	FIGURE 5: EVOLUTION DES CAPTEURS	10
FIGURE 2- 2: ESTIMATION DE POSITION PAR TRILATERATION	CHAPITRE 2	
CHAPITRE 3  FIGURE 3- 1STRUCTURE CORPORELLE DE LA DROSOPHILE	FIGURE2- 2: ESTIMATION DE POSITION PAR TRILATERATION	23
FIGURE 3- 1STRUCTURE CORPORELLE DE LA DROSOPHILE		
FIGURE 3- 2SCHEMA DE RECHERCHE DE LA DROSOPHILE	CHAITIRE 5	
FIGURE 4-1 L'INTERFACE DU MATLAB	FIGURE 3-2SCHEMA DE RECHERCHE DE LA DROSOPHILE.	32
FIGURE 4- 2 LA COORDONE ESTIME DE NOEUD INCONNU PAR CENTROID	CHAPITRE 4	
FIGURE 4- 2 LA COORDONE ESTIME DE NOEUD INCONNU PAR CENTROID		
FIGURE 4-3 L'ORGANIGRAMME PROPOSE		
FIGURE 4- 4 LES ETAPES DE SIMULATION		
FIGURE 4- 5 L'INTERFACE DE SIMULATION.       4         FIGURE 4- 6 LA LOCALISATION AVEC L'ALGORITHM PROPOSE.       4         FIGURE 4- 7 ERREURE DE LOCALISATION PAR RAPPORT NOMBRE DE CAPTEURS.       4         FIGURE 4- 8 ERREURE DE LOCALISATION PAR RAPPORT RAYON DE CONNECTIVITE.       4		
FIGURE 4- 6 LA LOCALISATION AVEC L'ALGORITHM PROPOSE       42         FIGURE 4- 7 ERREURE DE LOCALISATION PAR RAPPORT NOMBRE DE CAPTEURS       44         FIGURE 4- 8 ERREURE DE LOCALISATION PAR RAPPORT RAYON DE CONNECTIVITE       44		
FIGURE 4-7 ERREURE DE LOCALISATION PAR RAPPORT NOMBRE DE CAPTEURS		
FIGURE 4-8 ERREURE DE LOCALISATION PAR RAPPORT RAYON DE CONNECTIVITE		
FIGURE 4-9 ERR DE LOCALISATION PAR RAPPORT NOMBRE DES ANCRES		
FIGURE 4- 10 ERREURE DE LOCALISATION PAR RAPPORT LA TAILLE DE POPULATION45		
FIGURE 4-11 ERREURE DE LOCALISATION PAR RAPPORT NOMBRE D'ITERATION		46
FIGURE 4- 12 LA COMPARAISON ENTRE L'ALGORITHMES CSO AMELIORE,CSO ET L'ALGORITHME PROPOSE	•	47

#### LISTE DES TABLEAUX

TABLEAU 2- 1: COMPARAISON DES METHODES DE DERIVATION DE POSITIONS	25
TABLEAU 2- 2: COMPARAISON DES TECHNIQUES D'ESTIMATION DES DISTANCES	26
TABLEAU 2-3: CARACTERISTIQUES DES ALGORITHMES DE LOCALISATION DANS LES RCSFS	28
TABLEAU 4- 1 LES PARAMETRES D'EVALUATION	43
TABLEAU 4- 2 LES PARAMETRES D'EVALUATION	46
TABLEAU 4-3 ERR DE LOCALISATION PAR RAPPORT LE NOMBRE D'ANCRES	47

#### Introduction générale

Les réseaux de capteurs provoquent un intérêt croissant au sein des communautés scientifiques et industrielles depuis plusieurs années. Concrètement, un réseau de capteurs sans fil est composé d'un ensemble de capteurs intelligents qui sont capables de récolter et traiter des informations, et ayant aussi la capacité de communiquer entre eux. C'est ainsi que les réseaux de capteurs sans fil ont provoqué un intérêt important dans le monde scientifique se traduisant par une avancée technologique considérable notamment dans les domaines militaire, médical, environnemental etc....

Les perspectives d'utilisation des RCSFs sont claires et attractives, mais les problèmes générés par ces réseaux sont encore nombreux. Initialement, il ne dépend d'aucune infrastructure et les capteurs ne contiennent aucune information sur le réseau auquel ils appartiennent. De plus, étant construits sur une base ad hoc, ces réseaux se caractérisent par l'auto-organisation.

Parmi les défis des RCSFs, le problème de la localisation est critique : il s'agit d'attribuer une localisation géographique (précise ou estimée) aux capteurs ; Une application telle que la surveillance des feux de forêt, par exemple, n'aura aucun sens sans connaître la position de l'évenment. L'emplacement est donc un problème majeur dans de nombreuses applications où connaître l'emplacement du capteur fournit des informations très utiles. La localisation est devenue une information à grande valeur ajoutée, tout comme la sécurité ou la logistique, de plus, il est souvent souhaitable de connaître la localisation des capteurs dans l'environnement, afin de pouvoir déterminer l'origine des flux mesures.

Le système GPS est le système de localisation le plus utilisé au monde. Cependant, il revient couteux d'équiper tous les nœuds du réseau par un diapositive GPS. D'autres solutions ont été proposées des algorithmes de localisation (auto localisation).

Récemment, un algorithme inspiré de la nature a attiré l'attention des chercheurs en raison de sa faible complexité et de sa capacité à produire une solution avec les ressources limitées de RCSF. La plupart des algorithmes inspirés de la nature sont des algorithmes stochastiques, qui incluent de nombreux paramètres aléatoires dans leur procédure. Les paramètres permettent aux algorithmes de couvrir plusieurs zones de la zone de recherche simultanément, ce qui augmente les chances de ne pas être piégé au niveau d'un optimum local.

Dans cette mémoire, nous avons cherché à implémenter de nouveaux algorithmes distribués utilisant des méta-heuristiques bio-inspirées pour apporter des réponses à de nombreux problèmes posés par la localisation dans les RCSF. Nous avons accordé une grande attention à l'amélioration de la localisation. Pour cela, nous utiliserons une nouvelle approche hybride entre deux méta-heuristique une s'appelle l'algorithme d'optimisation par fruit Fly (FOA), et l'autre c'est l'optimisation par un essaim de particule (PSO). Nous avons utilisé le simulateur MATLAB, pour implémenter notre proposition pour la localisation. Ensuite, nous avons mené une étude analytique comparative afin de prouver son succès dans un réseau statique 2D.

Pour faire un processus, nous l'avons nettoyé en quatre chapitres comme ci-dessous

•

Le chapitre 1 : nous présentons Etat de l'art sur les réseaux de capteurs sans fil. Le chapitre 2 : est consacré à étude des algorithmes de localisation dans les réseaux de capteurs sans fil.

Le chapitre 3 : nous décrivons les méta-heuristiques d'optimisation PSO et FOA. Le chapitre 4 : constitue le cœur de notre travail, dans ce chapitre nous présentons l'application et les résultats de simulation obtenu.

Nous terminerons ce mémoire par une conclusion générale est résumer de tous le travail.

## **CHAPITRE 1**

# RESEAU CAPTEUR SANS FIL RCSF

#### 1. <u>Définition</u>:

Les RCSF sont des systèmes embarqués. Ils sont constitués d'un groupe d'unités de traitement embarquées appelées « mottes » qui communiquent via des liaisons sans fil. Plusieurs unités sont déployées pour la collecte ou la saisie de données. Grâce aux récentes avancées technologiques, de très petits capteurs ont été développés, à faible coût et à très faible consommation d'énergie.

Afin de mieux le comprendre, la figure suivante représente son mode général de fonctionnement.

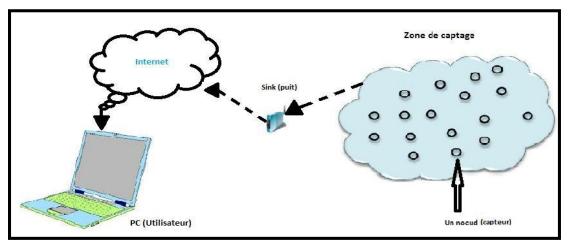


Figure 1: Architecture d'un RCSF

#### 2. <u>Les éléments constitutifs d'un RCSF:</u>

Le réseau de capteurs se compose d'un grand nombre de nœuds qui sont placés à un endroit spécifique ou de manière aléatoire.

Ce ispersement aléatoire des capteurs nécessite que le protocole utilisé pour les réseaux de capteurs possède des algorithmes d'auto organisation.

De ce fait il est primordial que la durée de vie des batteries soit la plus longue possible, Il a donc été nécessaire d'allier technologie et low-cost.

Les capteurs peuvent être programmés à diverses fins, telles que le contrôle des intrusions, le calcul de la température, le changement climatique et la surveillance. des déplacements d'animaux, surveillance de malades

Un réseau de capteurs sans fil se compose généralement de plusieurs nœuds Capteurs, un nœud, Sink et un centre de traitement de données. [2]

- **Nœuds :** Ce sont des capteurs qui répondent aux exigences de l'application pour laquelle ils sont conçus du point de vue de l'architecture et de la dispersion géographique.
- Sink ou puits : Il s'agit d'un nœud important du réseau. Sa tâche est de collecter des données provenant d'autres nœuds du réseau.
- Un centre de traitement de données: Toutes les données collectées par le Sink y seront envoyées, ce centre va regrouper toutes les informations et les triées afin de relever celles qui seront exploitables.

#### 3. Qu'est-ce qu'un capteur :

#### 3.1. <u>Définition</u>:

Les capteurs sont des dispositifs miniaturisés possédants des ressources énergétiques limités et autonomes, capables de traiter des informations et de les transmettre via des ondes radio [3]. Parmi ces phénomènes récoltés nous pouvons citer ceux ayant trait à la température, l'humidité, la pression, capture d'image, etc...

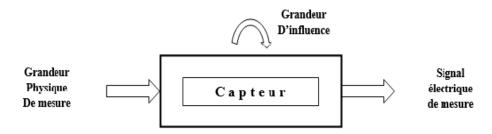


Figure 2: Exemple de capteur sans fil

En plus de ces capacités de multimètre, les capteurs existants ont des fonctionnalités de gestion expérimentées qui leur permettent d'enregistrer et de détecter des événements mesurables., le traitement de ces données et leur communication vers un autre dispositif [4]. On parle alors de capteur intelligent, capable à la fois de mesurer des données, stocker et les communiquer avec d'autres capteurs au sein d'un réseau, tel qu'il est caractérisé par grâce à ses capacités à collecter, traiter et communiquer les mesures au monde extérieur.

#### 3.2. Architecteur et composition d'un capteur sans fil :

Selon l'application : le capteur comprend : la détection, le traitement, la communication et l'alimentation. Des composants supplémentaires peuvent être ajoutés en fonction du domaine d'application, par exemple un système de localisation environnementale tel que le GPS et un système de navigation.

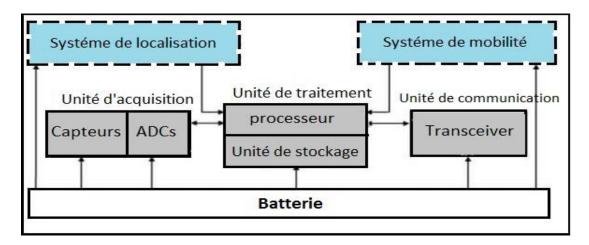


Figure 3: Architecture matérielle d'un capteur sans fil

#### • L'unité d'acquisition

Il se compose généralement de deux sous-unités, qui sont les capteurs et les ADC (Analog Digital Inverter). Les capteurs permettent de mesurer des paramètres environnementaux pour fournir des signaux analogiques obtenus après une conversation de ces données collectées. De plus, vous devez en savoir plus.

Les CAN convertissent ces signaux analogiques en signaux numériques.

#### • L'unité de traitement

Il est divisé en deux parties, l'acquisition et la communication, et son rôle est de contrôler le bon fonctionnement des autres unités.

Cette unité permet la mise en œuvre de procédures de communication qui permettent à un nœud de collaborer avec d'autres nœuds du réseau ; Il permet également d'analyser les données collectées afin d'atténuer l'action du nœud pelvien.

#### • L'unité de communication

Cette unité permet d'effectuer toutes les communications entre les différents nœuds sur un médium sans fil, car elle est dotée d'un émetteur/récepteur.

#### • <u>Batterie</u>

Il équipe les unités que nous avons mentionnées et n'est généralement ni rechargeable ni remplaçable. C'est un inconvénient majeur lors de la conception de protocoles pour les réseaux de capteurs.

Il existe des capteurs qui contiennent d'autres composants supplémentaires tels que les systèmes de suivi GPS.

#### • La mobilité

Un mobilisateur peut parfois être nécessaire pour déplacer un nœud pour accomplir ses tâches, le mobilisateur peut également opérer dans l'interaction étroite avec l'unité de détection et le processeur pour contrôler les mouvements du nœud. [5].

#### 3.3. Exemples des types de capteurs :

Il existe actuellement différents types de capteurs commercialisés sur le marché, avec des caractéristiques et des fonctions différentes.

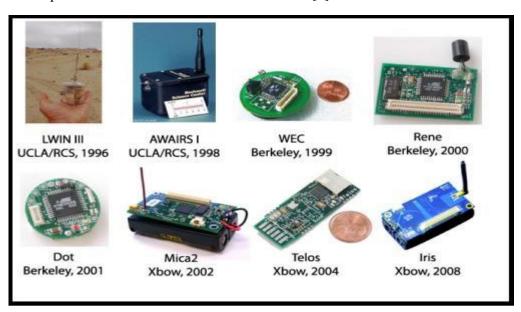
Sachant que la grande majorité des capteurs sont conçus en fonction de l'application pour laquelle ils sont utilisés, cela nous donne une grande variété de capteurs.



Figure 4: Schéma représentant quelques types de capteurs

Les capteurs ont connu une évolution fulgurante au fil du temps, et il est intéressant de voir cette évolution au cours des 20 dernières années. On note que la société a la distinction et se démarque des autres, il s'agit de Xbow également appelé Cross Bow.

Nul doute que les capteurs fabriqués par Xbow au cours des dix dernières années (famille de capteurs Mica, Telos, etc.) sont les plus largement utilisés dans les expériences et la recherche. Ces capteurs sont capables de mesurer plusieurs métriques (température, humidité, etc.) et la plupart tournent autour du Chipcon CC2420 qui est devenu le standard des émetteurs utilisant le protocole de communication IEEE 802.15.4 [6].



**Figure 5**: Evolution des capteurs

#### 4. Les réseaux de capteurs sans fil

#### 4.1. <u>Définition d'un RCSF</u>

Un réseau de capteurs peut être considéré comme un réseau de microsystèmes indépendants répartis sur une zone de rassemblement ou une zone d'intérêt et qui communiquent entre eux viaune liaison sans fil. Les éléments de ce réseau consistent en un nombre très important de capteurs capables de récolter et de transmettre des données environnementales au nœud passerelle (puits ou en anglais « sink ») d'une manière autonome, et mettant en œuvre un routage jusqu'au nœud considéré comme un point de collecte [7]. (Voire la figure 1).

#### 4.2. Le déploiement des nœuds du RCSF

Le déploiement est la première étape de la mise en place d'un réseau de capteurs sans fil. C'est une étape, tout comme l'analyse des informations collectées par la station de base, qui nécessite une intervention humaine [8]. Cette étape peut se faire de deux manières :

- **Déterministe** : Cela signifie que l'emplacement des capteurs est bien défini, c'est-à-dire qu'il est connu à l'avance..
- **Aléatoire** : dans ce cas, les capteurs s'auto organisent et utilisent des stratégies de localisation pour former le réseau.

#### 4.3. Caractéristique des RCSF:

Cependant, les caractéristiques importantes du RCSF sont les suivantes :

- Évolutivité et adaptabilité : L'importance du nombre de nœuds qui peuvent parfois atteindre des millions pour une meilleure précision de la surveillance, avec une variété de substitutions en cas de panne.
  - Caractéristiques du trafic interrompu : L'accès sans fil est bloqué en raison d'une accumulation à l'intérieur du RCSF (Exp : perturbation des liens radio dans un hôpital).
  - **Réseaux multi-sauts :** Les RCSFs est basé sur la communication multi-sauts en raison des limitations des ressources physiques dues à la miniaturisation des matériaux électroniques
- **Réduction de la consommation de l'énergie :** La gestion de l'énergie au sein du RCSF est limitée en raison du fait que les capteurs sont alimentés par des batteries qui ne peuvent pas être changées en raison d'environnements hexadécimaux défavorables.
- **Différents types de déploiements** : Engendrent des conséquences qui diffèrent d'un déploiement à un autre :
- o Le déploiement dans la nature mène à une absence de sécurité physique causée par la présence d'intrus ou de capture de nœuds.
  - o Le déploiement dans un lieu précis exige une topologie préconfiguré.
  - o Le déploiement aléatoire demande des algorithmes d'auto organisation.

#### 4.4. Cas d'application des RCSF :

Les capteurs peuvent être programmés à diverses fins, telles que le contrôle des intrusions, le calcul de la température, le changement climatique, la surveillance, des déplacements d'animaux (avec récepteurs GPS), surveillance de malades, ... [2].

Les cas d'application étant fort nombreux ; nous citerons à titre d'exemple :

#### **4.4.1.** <u>Feux de forêt :</u>

La pose de capteurs aide à prévenir en temps utile les risques d'incendie de forêt en déclenchant une intervention de secours ou de prévention.

#### 4.4.2. <u>Domaine militaire</u>:

Pour cette institution, un réseau de capteurs offre des avantages très précieux, dans le contrôle des équipements et des munitions, dans la communication à faible coût entre les unités avec une logistique, une reconnaissance et une surveillance simples dans le domaine du tir.

Il existe également de nombreux domaines qui utilisent le réseau de capteurs sans fil.

#### 4.5. Les limites des RCSF:

La localisation dans les réseaux de capteurs comporte certaines limites et ceci causés par l'absence de dispositif d'auto positionnement.

En raison de la diversité des limitations de conception, nous mentionnons les plus importantes :

- Confirmation de la continuité du réseau sans erreur.
- La mémoire a la capacité de stocker toutes les informations reçues.
- Maitrise des couts de production.
- Surmonter les défaillances liées à l'environnement.
- Répartition topologique grâce à des modifications assidues.
- Cornement du facteur matériel telles que la taille des capteurs le gain d'énergie, l'adaptation aux milieux et sa résistance.
- Mise en place d'une norme de transmission telle que l'infrarouge, le Bluetooth et les communications radio Zig Bee [9].
- Une adéquation entre l'utilisation et la consommation de l'énergie capable d'assurer une longévité du réseau.

#### Conclusion

A travers ce chapitre, on note une large présentation des réseaux de capteurs sans fil dans leur polyvalence ainsi que leur mode de fonctionnement, et on est plus clair par la suite étude de leurs caractéristiques et leurs spécificités. Des détails sur les capteurs et leur utilisation sont également abordés dans ce chapitre.

## **CHAPITRE 2**

# La localisation dans les RCSF

#### 1. Introduction

Connaître la position d'un dispositif dans son environnement est une question importante pour les applications offrant des services basés sur la localisation.

La localisation des nœuds est très nécessaire car elle est la première tâche exécutée après le déploiement des nœuds et après la détection d'un événement la question qui se pose et où se trouve cet événement !!!, donc c'est ça l'utilité de la localisation.

Les systèmes de localisation automatisés actuels sont divers et utilisent de nombreuses voies technologiques, ils sont très différents les uns des autres et répondent à de nombreux besoins.

#### 2. <u>Définition d'un système de localisation :</u>

La localisation est une méthode de placement d'un objet sur une carte ou une carte géographique, et cette méthode est mise en œuvre par un terminal dont la localisation peut être connue en temps réel ou en différé. Les positions enregistrées peuvent être stockées au sein du terminal et être extraites postérieurement, ou être transmises en temps réel vers une plateforme logicielle de localisation.

La transmission temps réel nécessite un terminal équipé d'un moyen de télécommunication qui permet d'envoyer les positions à des intervalles réguliers [10].

#### 3. Système de localisation :

#### 3.1 Critères d'un système de localisation :

L'algorithme de localisation est évalué selon les critères suivants :

#### 3.1.1 Précision de la localisation :

Nous parlons de l'erreur qu'il y a entre les vraies positions des nœuds et les positions calculées par la localisation. [11]

#### 3.1.2 <u>Contraintes de ressources :</u>

Les nœuds capteurs ont généralement des ressources très limitées. Ils ont une petite mémoire et des processeurs faibles, ce qui rend les calculs volumineux peu pratiques. Par conséquent, l'algorithme de localisation doit être simple et non compliqué, et son développement ne nécessite pas de grandes opérations arithmétiques ou une grande mémoire. De plus, nous ajoutons la rapidité de l'algorithme. [12]

#### 3.1.3 Contraintes énergétiques :

Le nœud de capteur tire son énergie de la batterie. Pour cela dans les réseaux de capteurs, il est nécessaire de préparer une alimentation très économique. Comme le facteur dominant de la consommation d'énergie est la communication radio, il faut trouver un algorithme de localisation qui communique le moins possible via la radio [12].

#### 3.1.4 Passage à l'échelle :

Un algorithme a-t-il la capacité de fonctionner sur un réseau de plusieurs milliers de nœuds ? Et si oui, est-il toujours aussi efficace. Ce critère est en rapport avec le fait qu'un algorithme soit imprésentable de façon distribuée ou non. [11]

#### 4. La localisation dans les RCSF

Souvent, les emplacements des capteurs n'ont pas besoin d'être connus dans l'environnement, afin de pouvoir déterminer l'origine du flux de mesures collectées. Les méthodes de localisation étudiées dans ce mémoire ont pour but d'estimer ces positions de manière automatique.

Une méthode de localisation dans les réseaux de senseurs est composée de deux parties :

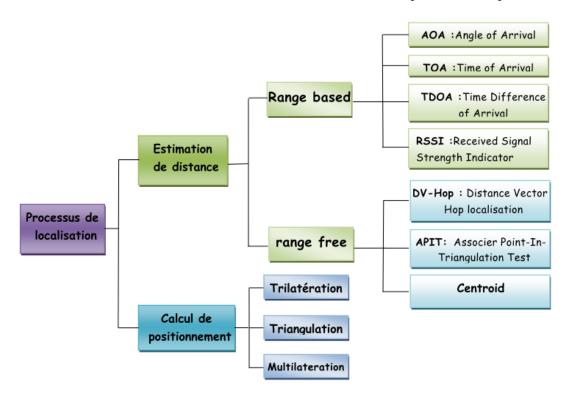


Figure 2- 1: Processus de localisation

#### 4.1 Estimation des distances

Dans cette phase les nœuds communiquent entre eux et collectent différents indicateurs de qualité des communication radios. La radio peut exprimer diverses informations sur le signal radio entre deux nœuds.

En fait, le fait qu'ils communiquent ensemble nous indique qu'ils sont à portée radio l'un de l'autre. De plus le hardware radio de nos nœuds peut nous rapporter diverses caractéristiques à propos du signal radio entre les deux nœuds, à partir desquelles les distance séparant les nœuds peuvent être estimées.

Le problème d'estimation des distances sera étudié en détail. [13]

#### **4.2** <u>Dérivation des positions</u>

Le but de cette phase est de trouver les positions des nœuds qui respectent au mieux les distances internoeuds estimées.

Si nous connaissons l'emplacement de certains nœuds du réseau dans un système de coordonnées particulier, alors l'emplacement d'autres nœuds dans ce système de coordonnées peut être déterminé.

Le problème de la dérivation des positions sera étudié plus en détail.

Notons que divers raffinements peuvent être appliqués, par exemple : Les distances estimées entre les nœuds peuvent être raffinées en refaisant communiquer les nœuds. De la même façon, on peut affiner la position des nœuds et appliquer diverses vérifications sur leurs positions. Une dernière transformation des positions de nœuds peut être réalisée afin de les intégrer dans une carte existante.

#### 5. Techniques de mesure :

La localisation dans les WSN's repose sur des mesures. Il y a plusieurs facteurs qui influent sur le choix de l'algorithme à utiliser pour une application spécifique et la précision de l'emplacement estimé. Les calculs définissent le type d'algorithme qu'un système de suivi particulier peut utiliser. Nous identifierons ensuite les différentes méthodes d'exploitation ou nécessaires des deux techniques situationnelles.

#### 5.1. Mécanismes de dérivation des positions :

Il existe deux principaux types de méthodes d'ingénierie à notre disposition pour utiliser ces informations, à savoir :

#### 5.1.1. <u>La trilatération :</u>

Cette méthode nécessite la connaissance de la distance entre le capteur inconnu et chaque nœud d'ancrage [14].

Soient trois nœuds d'ancrages B1, B2 et B3, M est le nœud que nous voulons localiser. Les positions (xi, y i) des balises ainsi que les distances Di sont connues. La relation entre M, B1 et B2 peut être écrit comme :

$$(x + x_1)^2 + (y + y_1)^2 = d_1^2$$
 (2,1)

$$(x + x_2)^2 + (y + y_2)^2 = d_2^1$$
 (2,2)

M est donc situé à l'intersection des cercles C1 (B1, d1) et C2 (B2, d2). Un troisième cercle généré en utilisant les informations de B3 nous permet de choisir la position correcte entre les deux solutions du cas général de deux cercles entrecroisés

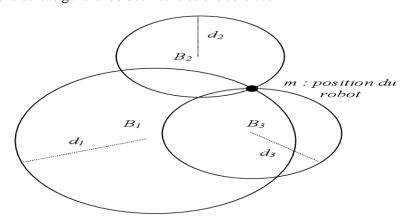


Figure 2- 2: Estimation de position par trilatération

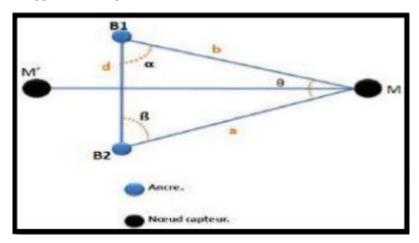
#### 5.1.2. La triangulation

Afin d'appliquer cette méthode, le nœud capteur inconnue doit être conscient de la distance entre les deux ancres et les angles et comme indiqué sur la Figure II.3. La loi des rendements sinus :

$$\alpha = \frac{d \times \sin \alpha}{\sin(\pi - \alpha - \beta)}$$
 (2,3)

$$\beta = \frac{d \times \sin \beta}{\sin(\pi - \alpha - \beta)}$$
 (2,4)

Une fois que les caractéristiques du triangle sont connues, les coordonnées du troisième sommet peuvent être calculées. Comme indiqué en décrivant trilatération, troisième ancre ou une hypothèse supplémentaire permettra l'identification de la solution correcte [15].



**Figure 2- 3:** Estimation de position par Triangulation.

Le point M se retrouve à l'intersection des droites passant par les couples (B1, M) et (B2, M) Figure II.3. En définissant un nouveau repère dont l'origine est B1 où (B1, B2) se confond avec l'axe des ordonnées.

#### 5.1.3. <u>Multilatération</u>:

Le multilatéralisme est le cas le plus fondamental et le plus axiomatique. Cette méthode calcule la position d'un nœud par l'intersection de cercles. Pour estimer sa position, un nœud a besoin de connaître la position d'un certain nombre d'ancres ainsi que sa distance par rapport à ces ancres.

Soit une cible 'a' dont on veut trouver la position  $x_a$ , et soit 'm' ancres i dont nous connaissons les positions  $x_i \in \mathbb{R}^p$ ,  $1 \le i \le m$ .

Nous supposons que nous connaissons aussi une estimation des distances  $\hat{d}$ ,  $1 \le i \le m$ entre chaque ancre i et le nœud a. Nous pouvons alors poser :

$$(x_{11} - x_{\alpha 1})^2 + (x_{12} - x_{\alpha 2})^2 + \dots + (x_{1p} - x_{\alpha p})^2 = \widehat{d}_{1\alpha}^2$$
 (2,5)

$$(x_{m1}-x_{\alpha 1})^2+(x_{m2}-x_{\alpha 2})^2+\cdots+(x_{mp}-x_{\alpha p})^2=\widehat{d}_{m\alpha}^2 \qquad (2,6)$$

Le système peut être linéarisé en soustrayant la dernière équation des m-1 équations précédentes

$$A = \begin{bmatrix} 2(x_{11} - x_{m1}) & \cdots & 2(x_{1p} - x_{mp}) \\ \vdots & \ddots & \vdots \\ 2(x_{(m-1)1} - x_{m1}) & \cdots & 2(x_{(m-1)p} - x_{mp}) \end{bmatrix}$$
(2,7)  

$$b = \begin{bmatrix} x_{11}^2 - x_{m1}^2 + \cdots + x_{1p}^2 - x_{mp}^2 + d_{ma}^2 - d_{ma}^2 \\ \vdots \\ x_{(m-1)1}^2 - x_{m1}^2 + \cdots + x_{(m-1)p}^2 - x_{mp}^2 + d_{ma}^2 - d_{(m-1)a}^2 \end{bmatrix}$$
(2,8)

Comme nous avons des erreurs dans les estimations de distances, nous ne pouvons pas trouver de solution exacte à ce système d'équations. La solution la plus proche (signifiant les moindres carrés) La solution exacte est connue par minimisation  $(Ax - b)^T (Ax - b)$  [16]

Et donc

$$\hat{\mathbf{x}}_{\mathbf{a}} = \operatorname{argmin}_{\mathbf{x}} (\mathbf{A}\mathbf{x} - \mathbf{b})^{\mathsf{T}} (\mathbf{A}\mathbf{x} - \mathbf{b}) = (\mathbf{A}^{\mathsf{T}}\mathbf{A})^{-1} \mathbf{A}^{\mathsf{T}}\mathbf{b}.$$
 (2,9)

$$\widehat{\mathbf{X}}_{\mathbf{a}} = \begin{bmatrix} \widehat{\mathbf{x}}_{\mathbf{a}1} \\ \widehat{\mathbf{x}}_{\mathbf{a}2} \\ \vdots \\ \widehat{\mathbf{x}}_{\mathbf{ap}} \end{bmatrix}$$
 (2,10)

C'est-à-dire notre estimation de la position du nœud a.

#### **Bilan sur les méthodes de dérivation de positions**

Les trois méthodes qui peuvent être utilisées pour le calcul de la position ont été exposées. Il est à noter que cette offre n'est pas exhaustive. Le choix de la méthode dépend des informations collectées et des ressources du processeur. Le Tableau II-2 compare les principales caractéristiques des trois méthodes présentées dans les paragraphes précédents. [17].

<u>Tableau 2-1:</u> Comparaison des méthodes de dérivation de positions

Méthode	Nombre de références	Distance	Angle	Complexité temporelle	Défis
Trilatération	3	Oui	Non	O(1)	Sensible aux imprécisions des distances
Triangulation	3	Non	Oui	O(1)	Besoin de matériel supplémentaire
Multilatération	$n \ge 3$	Oui	Non	O(n <sup>3</sup> )	Complexité du calcul

#### 5.2. Modèle d'estimation des distances :

Cette phase d'estimation est très importante car elle sera utilisée par la suite pour les deux autres phases du système, le calcul de la position et l'algorithme de localisation. Plusieurs modèles existants pour estimer la distance/angle. Certains sont précis mais coûteux, nécessite un matériel supplémentaire, énergie, et des ressources processeur. D'autres sont imprécis mais ils sont déjà disponibles sur la plupart des capteurs et qui ne nécessitent aucun matériel supplémentaire [17].

L'estimation de distance peut être faite sur différents indicateurs de mesure parmi lesquels, on distingue :

- ✓ **RSSI** (Received Signal StrengthIndicator) : La puissance du signal à la réception.
- ✓ **LQI** (Link QualityIndicator) : Le taux d'erreurs corrigées lors des transmissions.
- ✓ **ToA/TDoA** : temps de parcours.
- ✓ **AoA/ DoA** (Angle of Arrival / Direction of Arrival) : angle d'arrivée et la portée de communication (nombres de sauts entre nœuds).

#### **&** Bilan des techniques de mesures

Le tableau 2-2 résume les caractéristiques de la technique d'estimation de distance décrite dans ce chapitre.

Méthode	Précision	Distance Maximale	Matériel supplémentaire	Défis (challenges)
RSSI	2-4 m	La portée	Rien	Variations du RSSI, Interférences
TOA	2-3 cm	La portée	Rien	Synchronisation des nœuds
TDOA	2-3 cm	Quelques mètres (2-10 m)	Capteur à ultrason	La distance maximale
AOA	Quelques degrés (5°)	La portée	Un ensemble de récepteurs	Travailler avec des nœuds de petite taille

Tableau 2-2: Comparaison des techniques d'estimation des distances

#### 5.3. Un aperçu des algorithmes de localisation existants

Dans cette section, nous donnons un aperçu non exhaustif des algorithmes utilisé pour la localisation dans les réseaux de capteurs sans fil.

#### 5.3.1. <u>DV-HOP et DV-distance :</u>

C'est le schéma le plus basique, il utilise un échange de vecteur de distance afin que tous les nœuds du réseau parviennent à calculer la distance les séparant de l'ancre.

Chaque ancre maintient une table {Xi, Yi, hi), où (Xi, Yi) Chaque ancre calcule la distance de l'ancre. Chaque saut ancre calcule la distance le séparant des autres ancres dans le réseau. C'est la distance par saut qui va constituer l'information de correction pour tout le réseau.

Chaque nœud ancre calcule:

$$HopSiz = \frac{\sum \sqrt{(Xi - Xj)2 - (Yi - Yj)2}}{HIJ} \qquad i \neq j \qquad (2,11)$$

L'idée principale derrière l'algorithme de distance DV est de répartir la distance mesurée entre voisins plutôt que le nombre de sauts.

Depuis la proposition des algorithmes de DV-hop et DV-distance, de nombreux autres algorithmes fondés sur le même Principe have been proposés. Ils visent à améliorer les performances des algorithmes de base. [18], [19].

#### 5.3.2. **APIT**:

La méthode APIT est basée on le Point-In-Triangulation test. Ce test permet à un nœud de déterminer s'il se trouve ou non à l'intérieur d'un certain triangle.

S'il se rapproche/s'éloigne de tous les sommets à la fois, le mobile est à l'extérieur du triangle. [20]

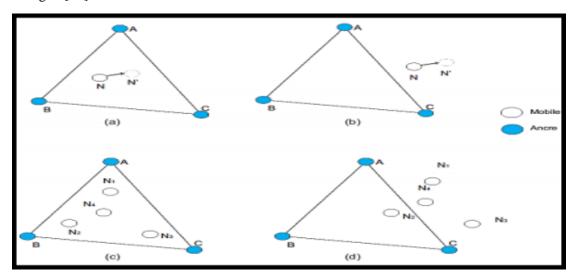


Figure 2-4: Principe de la méthode APIT

#### **5.3.3.** <u>Centroid</u>:

Cette méthode présentée dans [21], se base Essentiellement sur un certain nombre d'ancres. Un nœud voulant se localiser écoute pendant un certain temps les ancres. Il choisit à certaine nouvelle position, estime sa position and transmet un message 'beacon'

$$posi_{estim\acute{e}}(x,y) = \left[\frac{\sum_{i=1}^{n} x_i}{N}; \frac{\sum_{i=1}^{n} y_i}{N}\right] (2,12)$$

Avec,  $\sum_{i=1}^{n} x_i$ ,  $\sum_{i=1}^{n} y_i$  la somme des x et y des ancres écoutés par le nœud à localiser respectivement, et N le nombre de ces ancres.

Pour choisir les ancres, il calcule pour chaque ancre i entendu un coefficient CM<sub>i</sub>:

$$CM_i = \frac{nombre\ de\ message\ envoiyer\ par\ i}{nombre\ de\ message\ reçus} \times 100 \qquad (2,13)$$

#### Bilan sur les caractéristiques des algorithmes de localisation

Le Tableau II-3 résume les principales caractéristiques des méthodes de localisation présentées dans les paragraphes précédents.

<u>Tableau 2- 3:</u> Caractéristiques des algorithmes de localisation dans les RCSFs.

Caractéristique Méthode	Anchor- free	Anchor- based	Range-free	Range- based	Centralisé	Distribué
Bounding Box		#		#	*	#
DV-HOP		#	#		*	#
MDS-MAP	#			#	#	
APIT		#	#		*	#
SumDistMinMax		#	#		*	#
GPS-LESS		#	#		*	#
DV-Distance		#		#	*	#
DV-Euclidean		#		#	*	#

#### Conclusion

Dans ce chapitre, nous avons résolu le problème de la localisation des nœuds de capteurs dans un RCSF. Nous avons également introduit divers systèmes et techniques de positionnement pour estimer la distance entre deux nœuds. En plus de différents types d'algorithmes de positionnement, tels que la distance basée sur la distance et la distance libre. Ce dernier est plus populaire en raison de son coût inférieur et de sa facilité de mise en œuvre, et convient mieux à WSN. La précision et le faible coût sont des facteurs clés dans de nombreuses études, et les chercheurs y ont toujours un réel intérêt.

### **CHAPITRE 3**

# Optimisation par fruit Fly et Pso

#### 1. Introduction:

Les algorithmes d'intelligence en essaim sont une classe d'algorithmes méta-heuristiques auxquels on accorde de plus en plus d'attention au cours des trente dernières années. Ces algorithmes imitent le mécanisme d'échange d'informations, de partage d'informations et d'apprentissage parmi les animaux d'essaims en milieu naturel [22]. Le point commun de ces algorithmes est qu'ils obtiennent une solution de bonne qualité après un maximum d'itération.

Dans ce chapitre, Nous commençons par l'explication du problème d'optimisation, puis nous allons présenter les méta-heuristiques bio-inspiré et enfin, nous détaillerons le principe de fonctionnement des méta-heuristiques utilisées dans ce travail (PSO et FOA).

#### 2. Problème d'optimisation :

Le problème d'optimisation est défini comme l'optimisation de la recherche pour une fonction donnée. Les variables de cette fonction sont souvent contraintes d'évoluer dans une certaine partie de l'espace de recherche. On aura donc un problème d'optimisation sous contraintes. [23] Mathématiquement, on cherche à minimiser F sur E, c'est à dire, on cherche X\*E tel que :

$$f(x^*) = \min(ou \max)(f(x)) \tag{3.1}$$

#### 3. <u>Méta-heuristique d'optimisation :</u>

La méta-heuristique est un algorithme d'optimisation dont le but est de résoudre des problèmes d'optimisation difficiles pour lesquels on ne connaît pas de cas classique plus efficace.

Les métas heuristiques sont généralement des algorithmes stochastiques itératifs, qui progressent vers un optimum global, c'est-à-dire l'extremum global d'une fonction, par échantillonnage d'une fonction objectif. Ils agissent comme des algorithmes de recherche, essayant d'identifier les caractéristiques d'un problème afin de trouver une approximation de la meilleure solution (de la même manière que les algorithmes d'approximation).

Il existe de nombreuses heuristiques différentes, allant de la simple recherche locale aux algorithmes de recherche globale complexes. Ces méthodes utilisent cependant un haut niveau d'abstraction, leur permettant d'être adaptées à une large gamme de problèmes différents. [24]

#### 4. Optimisation par PSO:

L'optimisation par des essaims de particules est une technique informatique évolutive développée par Eberhart et Kennedy en 1995 [25]. Elle appartient à la classe des algorithmes d'optimisation stochastique qui simule le comportement social du vol d'oiseaux. L'algorithme PSO est facile à implémenter et il est efficace en termes de calcul car ses besoins en mémoire et en CPU sont faibles. [26]

La technique PSO utilise un ensemble de solutions réalisables appelées « particules » qui sont déployé dans l'espace de recherche avec des emplacements initiaux aléatoires. Les valeurs de la fonction objective correspondant aux emplacements des particules sont évaluées. Ensuite, les particules sont déplacées.

Dans l'espace de recherche en obéissant à des règles inspirées du comportement de vol des oiseaux [25].

Chaque particule est déplacée vers la moyenne pondérée de façon aléatoire de la meilleure position que la particule a rencontrée jusqu'à présent (Pbest) et de la meilleure position rencontrée par l'ensemble de la population de particules (gbest).

Soit Xi = (xi1, xi2, ..., xiM) le vecteur de N dimension représentant la position de la ième particule dans l'essaim, gbesti = [g1, g2, ..., gn] est le vecteur de position de la meilleure particule dans l'essaim (c'est-à-dire la particule avec la plus petite valeur de fonction objective). Pbesti = [pi1, pi2, ..., piN] est le vecteur de position de la ième particule et la vitesse de la particule  $Vi = [vi1. \ vi2, ..., \ viN]$  soit la vitesse de la ième particule. Les particules évoluent selon les équations :

$$P_{i,j}(t+1) = P_{i,j}(t) + V_{i,j}(t+1)$$
(3,2)

$$V_{i,j} = k[w * V_{i,j}(t) + c_1 * r_1 * (I_{i,j} - V_{i,j}(t)) + c_2 * r_2 * (g_i - V_{i,j}(t))]$$
(3,3)

Avec:

$$\frac{2}{|2-\varphi-\sqrt{\varphi^2-4\varphi}|}$$
,  $\varphi = c_1 + c_2 et \ \varphi > 4$ ,  $avecj = 1, ..., N$ , (3,4)

w: est une constante appelée facteur d'inertie.

c1 et c2 : sont des constantes appelées coefficients d'accélération.

r1 et r2 : sont des nombres aléatoires uniformément distribués dans l'intervalle [0,1].

Si la vitesse calculée entraîne l'éjection d'une particule de l'espace de recherche, son piégeage n'est pas calculé. Compte tenu de la nouvelle vitesse, obtenue à partir de pbesti et gbesti.

#### 5. Optimisation par Fruit Fly:

#### 5.1. Aspect biologique:

La drosophile ou mouche des fruits (Fruit Fly en anglais) est un insecte qui préfère de vivre dans des régions tempérées et tropicales, nourrissent principalement de plantes et de fruits en décomposition. Son corps (**Figure 3-1**) contient des organes olfactifs pour permettre de capturer de manière sensible les odeurs sur des dizaines de kilomètres. Pour rechercher de la nourriture, cette insecte utilise un processus repose principalement sur le transfert mutuel d'informations d'odeurs pour effectuer la migration de l'ensemble de la population vers l'insecte à la meilleure odeur. [26]



Figure 3- 1Structure corporelle de la drosophile

#### Modèle artificielle :

En 2011, le professeur Taïwan "Wen-Tsao Pan", simule le comportement de recherche de nourriture par la drosophile comme un modèle d'optimisation appelé FOA. L'optimisation par Fruit Fly est un nouvel algorithme de calcul évolutionnaire inventé pour résoudre les problèmes d'optimisation globale. Par rapport à l'optimisation par des essaims de particules (PSO), l'algorithme génétique (GA) et à d'autres méta-heuristiques classiques d'optimisation, l'algorithme FOA est simple, facile à mettre en œuvre et peut atteindre rapidement une optimisation précise.

Dans le modèle artificiel de l'algorithme, la taille de la population est invariante et pour atteindre l'objectif recherché, le FOA utilise une fonction objective pour estimer la concentration d'odeur. Ensuite, l'essaim vole par vision vers la mouche des fruits la plus proche de la nourriture.

À chaque fois, elle obtient le meilleur emplacement de l'insecte où la concentration de l'odeur est la plus élevée. Comme le montre la **Figure 3-2**, cette figure montre un diagramme simplifié de l'évolution de la recherche par la drosophile.

La forme ovale dans le schéma représente la zone de recherche dans laquelle un groupe de mouches des fruits est déployé de manière aléatoire pour la traversée de l'espace de la solution. Les points P1, P2 et P3 du schéma indiquent des emplacements candidats obtenu à travers un vol de l'essaim à partir la position initiale, et P2 est le meilleur emplacement sélectionné pour la prochaine itération de recherche. Dans la nouvelle recherche P1', P2' et P3' sont les nouveaux candidats obtenus à partir du centré de l'essaim sur P2. Ce processus se répète jusqu'à ce que l'objectif de la recherche soit trouvé.

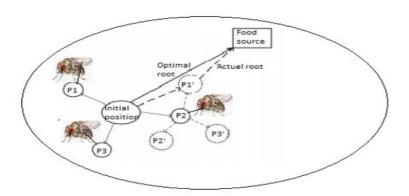


Figure 3- 2Schéma de recherche de la drosophile.

Selon les caracteristiques au groupe des moucnes des truits, les étapes de l'algorithme FOA sont décrites comme suite :

1. Initialisez aléatoirement l'emplacement initial du groupe de mouches des fruits, comme illustré sur la **Figure 3-3**.

#### Init (X\_axis, Y\_axis);

**2.** Recherchez la direction de la nourriture en donnant des valeurs aléatoires de la direction et à la distance de chacun des mouches des fruits.

Xi = X\_axis+RandomValue(); Yi = Y\_axis+RandomValue();

Où,  $i = 1 \dots$  Taille de la population.

**3.** Calculer la valeur de jugement de concentration (Si) c'est l'inverse de la distance à l'origine ; cela est calculée parce que l'emplacement de la nourriture ne peut être connu.

**Distance**<sub>i</sub> = 
$$\sqrt{X_i^2 + Y_i^2}$$
;  $S_i = \frac{1}{Distance_i}$ ; (3,5)

**4.** La valeur de concentration d'odeur (Si) est substituée dans la fonction de jugement de concentration d'odeur (fonction objective) pour déterminer la concentration d'odeur (Smelli) de la position de l'individu i.

Smelli = Function (Si);

**5.** Déterminer l'individu avec la concentration optimale (maximale ou minimale selon l'objective d'optimisation) d'odeur parmi l'ensemble des individus de la population.

[bestsmellbestindex]= optimal (smelli);

**6.** Conservez la meilleure valeur de concentration de l'odeur et leur coordonnées x, y. Les individus de l'essaim volent vers la position optimale par vision.

**Smellbest = bestSmell** 

 $X_axis = X (bestIndex)$ 

 $Y_axis = Y (bestIndex)$ 

**7.** Entrez dans une optimisation itérative, en répétez les étapes d'exécution 2 à 5 et jugez si la concentration d'odeur est meilleure que la concentration d'odeur de l'itération précédente, si oui, exécutez l'étape 6.

Les étapes détaillées de l'algorithme FOA sont présentées par l'organigramme de la **Figure 3-3** suivant :

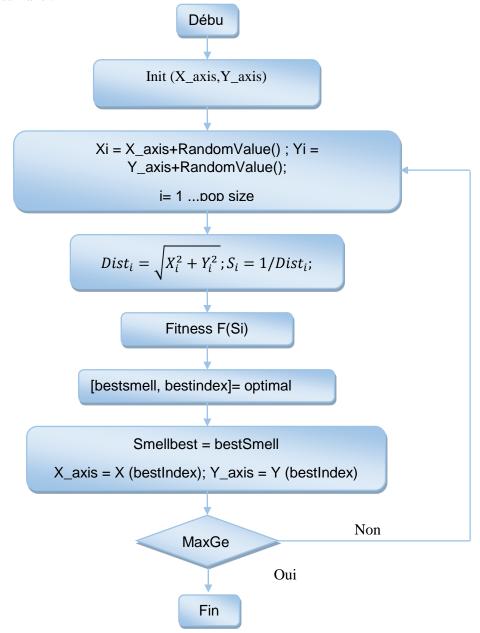


Figure 3- 3 Organigramme de l'algorithme FOA

#### **Conclusion:**

Au cours des dernières années, diverses méthodes d'optimisation méta-heuristique ont été développées. Dans ce chapitre, nous avons présentons deux algorithmes d'optimisation une c'est l'algorithme des essaims de particule et l'autre appelé L'optimisation par Fruit Fly, qui ont pour objectif de trouver les régions optimales dans un espace de recherche complexe grâce à l'interaction des individus de la population.

Dans le dernier chapitre qui ce suive nous présentons notre implémentons et évaluation de cette Algorithme d'optimisation dans La localisation des capteurs.

### **CHAPITRE 4**

# Implémentation et Discussion

#### 1. Introduction:

Dans ce dernier chapitre, on va présenter notre contribution pour la localisation dans les RCSFs. Notre approche proposée c'est une hybridation entre l'algorithme d'optimisation de Fruit Fly et la méta-heuristique des essaims de particules afin d'optimiser la précision de la localisation des nœuds dans les RCSFs. Pour mettre en œuvre notre proposition nous avons utilisé le langage de simulation MATLAB, et a fin de jugé la performance de Notre approche proposé dans la localisation leur résultats sont comparer avec ceux de la méta-heuristique PSO et CSO.

#### 2. Le langage de programmation utilisé :

Nous avons choisi MATLAB (MATrixLABoratory), comme un environnement d'implémentation pour notre application. MATLAB est un langage de calcul scientifique de haut niveau et un environnement interactif pour le développement des applications, la visualisation et l'analyse de données, ou encore le calcule numérique.

Nous utilisons MATLAB® 2016 pour résoudre des problèmes de calcul scientifique plus rapidement.

Le langage MATLAB met à la disposition du développeur les opérations vectorielles et matricielles, fondamentales pour les problèmes d'ingénierie et scientifiques.

Il permet un développement et une exécution rapide à l'égard de langage MATLAB nous pouvons programmer et tester des algorithmes plus rapidement qu'avec les langages traditionnels, car il n'est Pas besoin d'effectuer des tâches de bas niveau, exemple la spécification des types de données, la déclaration des variables et l'allocation de la mémoire

Nous avons développé notre application à l'aide du langage MATLAB version R2016a sur Windows 7 Professional 64 bit, de RAM de 4.00 Go, et de processus Intel(R) Core(TM) i3-3110M CPU @ 2.40 GHz, 2.40 GHz.

#### L'environnement de MATLAB possède 4 fenêtres

- Au centre l'invite de commande.
- En haut à droite : l'espace de travail.
- Gauche la liste des fichiers du répertoire courant.
- En bas à droite l'historique des commandes tapées
- Fenêtre GUI en MATLAB

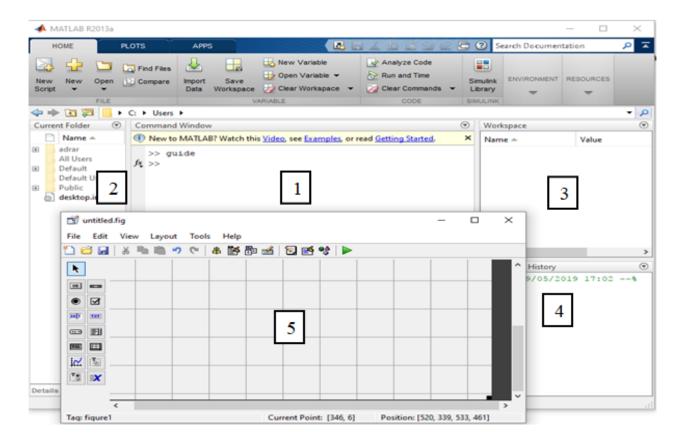


Figure 4- 1 L'interface du MATLAB.

# 3. Centroid:

Dans Notre approche proposé on utilisé l'algorithme centroid proposé par Bulusu et Heidemann, J., Estrin dans [27]. Cette méthode se base sur un quadrillage des nœuds de référence (ancres) émettant des message d'ancrage, pour calculer la position du nœud ordinaire ou inconnu qui ne connais pas sa position.

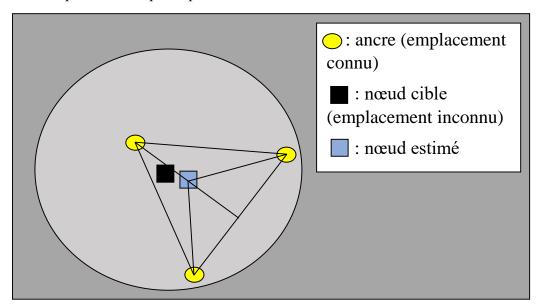


Figure 4-2 La coordoné estimé de Noeud inconnu par centroid

(X, Y): Les coordonnées du capteur qui ne connais pas sa position tel que :  $\frac{\sum_{i=1}^N x_i}{N}$ : la somme des x des ancres dans le rayon de connectivité du capteur i,  $\frac{\sum_{i=1}^N y_i}{N}$ : la somme des y des ancres dans le rayon de connectivité du capteur i et N: le nombre des ancres dans le rayon de connectivité du capteur i.

$$PE = \left[\frac{\sum_{i=1}^{N} x_i}{N}, \frac{\sum_{i=1}^{N} y_i}{N}\right]$$
 (4,1)

Les coordonnées des ancres dans le rayon de connectivité sont utilisées pour calculer le centroid de la zone et la position que le noeud considèrera comme la sienne. Sur la **Figure 4-2**, une réalisation de Centroid est illustrée avec uniquement les balises retenues représentées par un point. [28]

#### 4. Formulation de problème :

Le but de ce travail est de trouver l'emplacement de capteurs dispersés aléatoirement dans un plan 2D représente la zone d'intérêt de RCSF; Pour ce la nous avons considéré m>=0 nœuds appelés ancres dont ces positions sont connues a priori, donc l'objectif de la procédure de localisation est de trouver les positions de n nœuds inconnu étiquetés m+1, ... N (N=n+m). Chaque nœud de réseaux est caractérisé par un rayon de transmission R similaire.

L'estimation de position des nœuds cibles peut être formulée comme un problème d'optimisation, impliquant la minimisation d'un fitness représentant l'erreur dans la localisation des nœuds cibles. La somme des erreurs de distance au carré entre les nœuds cibles et les ancres voisins peut être considérée comme la fonction objective de ce problème [29]

Soit (x, y) les coordonnées du nœud ciblent et di soit la distance entre le nœud cible et le *ieme* ancre. La distance est calculée en utilisant la formule suivant

$$d_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}$$
 (4,2)

Soit  $\hat{d}_i = d_i + N_i$ la valeur de distance obtenue à partir la technique de mesure RSSI et $N_i$  c'est l'erreur commis lors de l'estimation de distance par RSSI. La fonction objective du problème de localisation est définie comme suite :

$$f(x,y) = \frac{1}{M} \sum_{j=1}^{M} |\hat{d}_j - d_j|$$
 (4,3)

- M > 3 Le nombre de nœuds de liaison dans le continuum du capteur cible
- (x', y') coordonne du nœud à estimer.
- (xj, yj) coordonne duj<sup>ieme</sup>ancre.
- di distance réel entre capteur inconnu et ancre.

#### 5. Description générale de l'approche de localisation proposé

Afin d'améliorer la précision de la localisation, on a contribué un nouvel algorithme hybride des deux méta-heuristiques FOA et PSO. Les étapes de notre proposition pour la localisation, sont décrites comme suit :

**Étape1**. Choix de la configuration de réseau et initialisation de paramètres de PSO et FOA.

**Étape 2.** Chaque ancre dans le réseau diffuse un message d'ancrage, permettre aux nœuds inconnu de se localiser.

Étape3. Trouver une place initiale pour chaque nœud inconnu en utilisons le Centroid.

**Étape4**. Évaluer les positions initiales calculées par centroid par la fonction objective et mettre sa valeur dans Gbest.

Étape5. Placer un group des fruits Fly dans la position initiale de chaque nœud inconnu.

**Étape6**. Utiliser les équations de mouvement de PSO pour trouver un nouvel emplacement au chaque individu de fruit Fly :

$$\mathbf{V}_{j}^{t+1} = \mathbf{V}_{j}^{t} + c_{1} * (1 * rand) * (pbestx_{j} + xx_{j}) + c_{1} * (1 * rand) * (gbestx_{j} + xx_{j})$$

$$(4,4)$$

$$\mathbf{T}_{j}^{t+1} = \mathbf{T}_{j}^{t} + c_{2} * (1 * rand) * (pbesty_{j} + yy_{j}) + c_{2} * (1 * rand) * (gbesty_{j} + yy_{j})$$

$$(4,5)$$

**Étape7**. évaluer les nouvelles positions par la fonction objective et mettre sa valeur dans Pbest.

**Étape8**. si un Pbest est inférieure au Gbest, mettre a jour le Gbest de nœud cible par cet valeur Pbest et faire regrouper tous les Fruits fly dans son emplacement.

**Étape9**. répéter les étapes 6, 7 et 8 jusqu'à un maximum d'itérations puis considéré le dernier emplacement de Gbest comme la position de nœud cible.

Étape10. Répéter les étapes 4.9 pour tous les nœuds inconnus.

Ces étapes de notre approche proposée sont illustrées dans l'organigramme suivant :

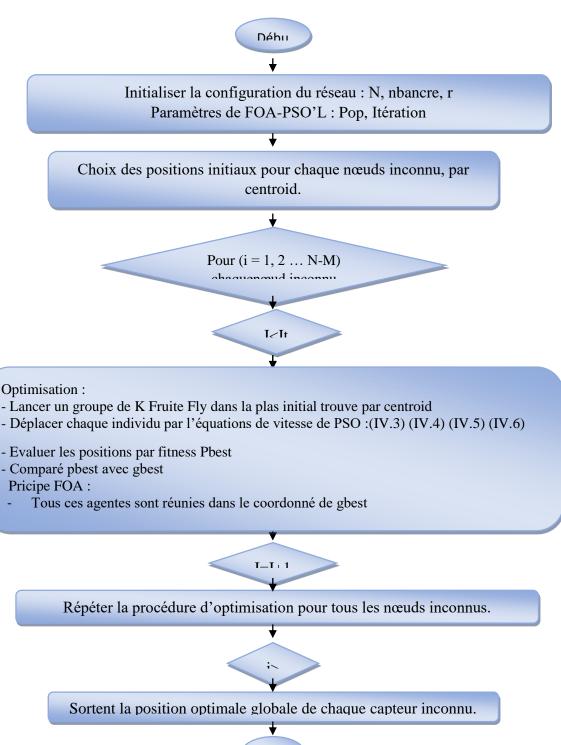


Figure 4- 3 L'organigramme propose.

Fin

# 6. <u>Les étapes de la simulation :</u>

Pour exécuter cette application, suivre les étapes suivantes :

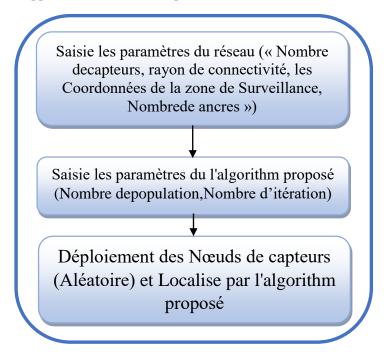


Figure 4- 4 Les étapes de simulation

# 7. <u>Description des étapes d'exécution d'application :</u>

L'interface principale de notre application est présentée dans la Figure 4-5

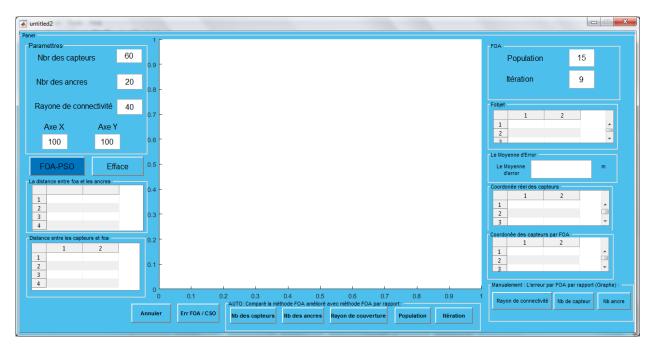


Figure 4-5 L'interface de simulation.

#### 7.1. Choix des paramètres de réseau :

Pour lancer l'algorithme, il faut d'abord saisir tous les paramètres initiaux suivant

- Nombre des capteurs :
- Nombre des ancres
- Le rayon de connectivite
- La zone de surveillance

# 7.2. Après de saisir les paramètres de réseau :

On a déployé aléatoirement les capteurs dans la zone de simulation par : un click sur le bouton de déploiement aléatoire.

## 7.3. Choix du paramètre de FOA et PSO:

Après de saisir tous les paramètres initiaux de réseau, on doit saisir les paramètres du l'algorithme proposé.

# 7.4. Localiser par l'algorithm proposé :

Localisation par l'algorithme proposé. Dans cette interface :

Les losangées oronges représentent les capteurs déployés aléatoirement, les losangés représentent les encres et Les cercles bleus indiquent les capteurs localisés.

Pour applique l'algorithme hybride FOA-PSO'L en clique sur le bouton « FOA-PSO »

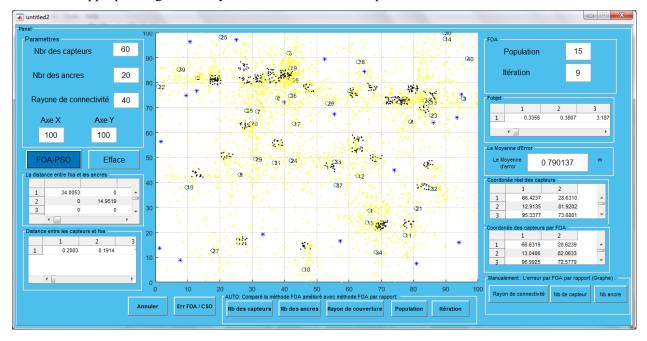


Figure 4- 6 La localisation avec l'algorithm proposé

L'étoile bleue indique les ancres, et le bleu les capteurs ne connaissant pas l'emplacement, les points noirs la propagation des éléments dans la première itération, et le bleu le reste de l'itération, Il est à noter qu'une très petite erreur a été obtenue avec le moins d'itérations et d'items.

# 7.5. L'Affichage des informations supplémentaire :

Pour affiche les informations en cliquant sur le bouton « Affichage », Ces informations sont

- a. Coordonnée réel des capteurs.
- b. Coordonnée calculé des capteurs.
- c. Distance entre les ancres.
- d. Distance entre les capteurs et les populations.
- e. N° Anchor lié au capteur.
- f. Fobjet: La meilleure valeur de fitness.

#### 8. Évaluation de l'algorithme proposé

Dans notre simulation, nous intéresserons essentiellement à la précision de localisation comme un critère d'évaluation puisqu'elle constitue une phase très importante qui permet de montrer l'efficacité de cette méthode de localisation.

#### 8.1. La précision de l'algorithme

Afin d'évalue la précision dans algorithme on peut calculer l'erreur moyenne produit par cette méthode. On doit calculer l'erreur moyenne comme étant la moyenne des différences entre les positions réelles et les positions estimées. L'erreur moyenne est calculée comme suit

Erreur = 
$$\frac{\sum \sqrt{(X_r - X_c)^2 + (Y_r - Y_c)^2}}{N}$$
 (4,6)

Tell que Xr, Yr: les coordonnés réels d'un noeud

Nombre d'itérations

Xc, Yc: les coordonnés d'un noeud trouvées par l'algorithem proposé.

N : nombre de capteurs

#### 8.2. Évaluation de notre application

Pour savoir l'effet les différentes paramétrées sur la précision de l'algorithme proposé, on a effectué la série d'expériences suivant :

Le tableau ci-dessus représente les paramètres fixés pour ces évaluations :

Les paramètres

zone de surveillance

100\*100m

Nombre des capteurs

60

Nombre des ancres

20

Rayon de connectivite

40

La taille de population

40

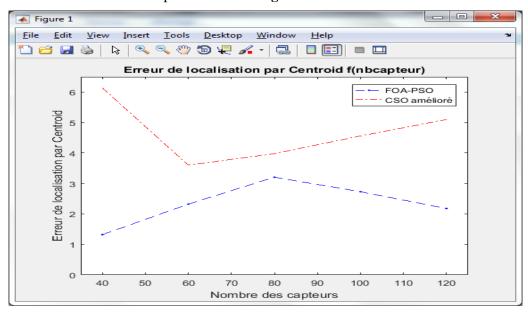
50

Tableau 4-1 Les paramètres d'évaluation

# 8.2.1. <u>La précision par rapport le nombre des capteurs :</u>

Dans cette expérience on fixant les différents paramètres sur les valeurs dans le **Tableau 4**-1et on variant le nombre de capteurs.

Les résultats obtenus sont présentés sur la Figure 4-7



**Figure 4- 7** Erreure de localisation par rapport nombre de capteurs

la **Figure 4- 7**montre clairement que la valeur de l'erreur moyenne est augmentée avec l'augmentation de nombre de capteurs dans les deux approches et on peut savoir aussi que les valeurs d'erreur moyenne de l'algorithm proposé est meilleure que CSO amélioré.

# 8.2.2. Précision par rapport au rayon de connectivite

De la même manière, l'erreur moyenne par rapport le rayon de couverture être comme ça.

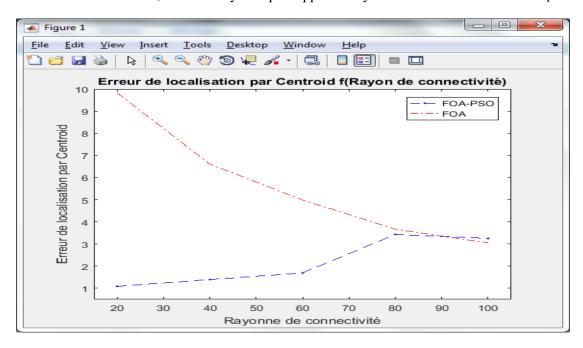


Figure 4-8 Erreure de localisation par rapport rayon de connectivite

D'après la **Figure 4-8**, on voit que l'erreur moyenne diminue avec l'augmentation de rayon de connectivite pour les deux algorithmes : CSO amélioré et l'algorithm proposé.

# 8.2.3. La précision par rapport aux nombre des ancres

Le graphe ci-dessous montre l'erreur de localisation par rapport le nombre des ancres :

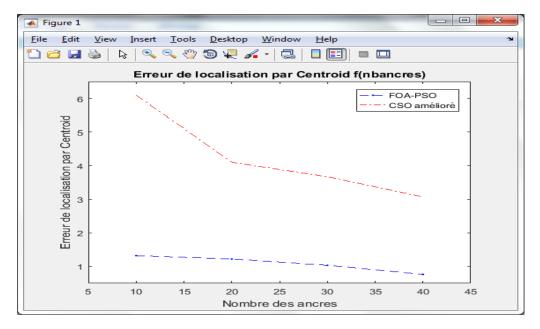


Figure 4- 9 Err de localisation par rapport nombre des ancres

Cette **Figure 4-9** montre clairement que la valeur de l'erreur moyenne diminue avec l'augmentation de nombre des ancres pour les deux algorithmes : l'algorithm proposé et CSO amélioré.

# 8.2.4. La précision par rapport la population :

Le graphe ci-dessous montre l'erreur de localisation par rapport la taille de population :

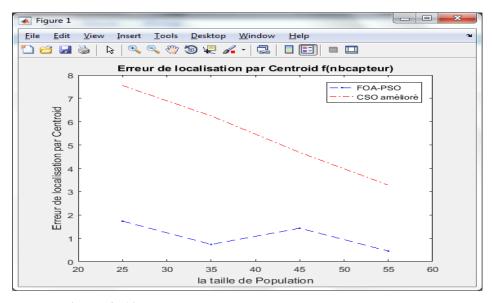


Figure 4- 10 Erreure de localisation par rapport la taille de Population

D'après la **Figure 4-10**, on voit que l'erreur moyenne diminue avec l'augmentation de la taille de population pour les deux algorithmes : l'algorithme proposé et CSO amélioré.et on voit aussi que l'erreur moyenne de l'algorithme proposé est plus petit par rapport l'algorithme FOA.

#### 8.2.5. La précision par rapport le nombre d'itération

Le graphe ci-dessous montre l'erreur de localisation par rapport le nombre d'itération :

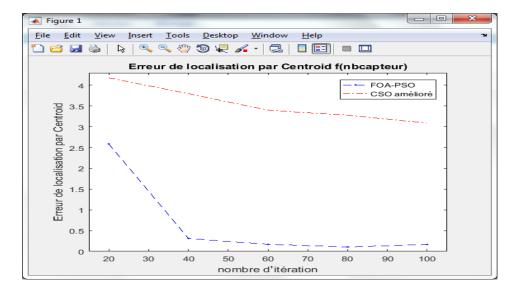


Figure 4- 11 Erreure de localisation par rapport nombre d'itération

Cette **Figure 4-11** montre clairement que l'erreur moyenne diminue avec l'augmentation le nombre d'itération pour les deux algorithmes : l'algorithme proposé et CSO amélioré.

Pour tester et vérifier les performances de notre proposition on comparer leur performance avec la méta-heuristique PSO présente dans [30]

Le tableau ci-dessus représente les paramètres fixes afin de trouver la différence :

Paramètres de réseaux		Paramètres de FOA		Paramètres de PSO	
Paramètres	valeurs	Paramètres	valeurs	Paramètres	valeurs
Zone de surveillance	100*100	Nombre d'itération	50	Nombre d'itération	50
Nombre des	100	Y . '11 1		W1	0.9
capteurs		La taille de population	0	W2	0.2
Rayon de	40	population			
connectivite				R1	2
				R2	2

Tableau 4- 2 Les paramètres d'évaluation

La **Figure 4-12** présente la comparaissant entre les trois algorithmes FOA-PSO, CSO améliorée et l'algorithme PSO

Le tableau ci-dessus représente l'erreur de localisation par rapport le nombre des ancres **Tableau 4- 3** Err de localisation par rapport le nombre d'ancres

Nombre des Ancres	20	15	20	25	30
Les algorithmes	,				
PSO	7	5.5	5.40	5.30	5.25
CSO améliore	2.36	2.3	2.36	2.2	2.05
l'algorithme proposé	2.36	2.3	2.36	2.2	2.05

A travers la **Figure 4-12** qui indique l'erreur de localisation par l'algorithme proposé, CSO amélioré et CSO par rapport le nombre des ancres, il est claire que la valeur de l'erreur de localisation est diminué avec l'augmentation de nombre de ancres, dans les trois algorithmes

On remarque que les valeurs d'erreur de localisation de l'algorithme CSO améliore est supérieur au algorithme CSO et l'algorithme proposé lorsque le nombre d'ancres égale 15, puis devient presque

Égal à celle de CSO. Et on peut voir aussi que la moyen d'erreur de l'algorithme proposé est inferieur par rapport les deux autres algorithmes

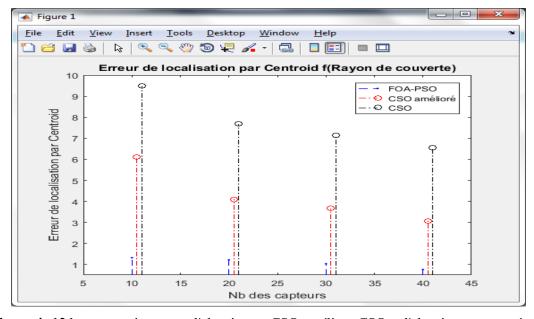


Figure 4- 12 la comparaison entre l'algorithmes CSO améliore, CSO et l'algorithme proposé

# **Conclusion:**

Notre simulation montre que l'augmentation de la taille des essaims améliore la précision de la localisation. En ajustant les paramètres du FOA avec les valeurs appropriées, nous pouvons obtenir le résultat optimal.

De plus, le modèle l'algorithme proposé est capable de fournir une solution d'optimisation complète avec le moins d'itérations. Afin d'évaluer notre algorithme, les résultats obtenus avec un FOA amélioré, dans la plupart des cas, fournissent de meilleurs résultats que CSO amélioré et la version standard de CSO.

Ce qui distingue cette méthode l'algorithme proposé est d'obtenir des résultats précis avec le moins d'itérations et de population.

# Conclusion générale

Les réseaux de capteurs sans fil ont un large potentiel et constituent un sujet de recherche innovant ainsi qu'un outil d'une grande utilité dans plusieurs domaines : militaire, médical, environnemental, surveillance, etc....

Cependant, il y a beaucoup de problèmes qui doivent être résolus dans ce cadre pour que nous puissions travailler avec eux dans de bonnes conditions. L'un des problèmes qu'on peut rencontrer dans ce genre de réseaux nous citons la problématique de la localisation.

Le thème de notre thèse est l'optimisation de la localisation en utilisent les métaheuristiques bio-inspirées. L'objectif visé est à la fois, l'adaptation des méta-heuristiques pour l'optimisation des techniques de localisation existantes ainsi que la proposition de nouveaux algorithmes de localisation.

Les méthodes méta-heuristiques peuvent apporter un plus à des problèmes d'optimisation d'une part et d'améliorer les performances de la localisation d'autre part.

notre approche de localisation est basé sur l'algorithme d'optimisation de la drosophile (ie, FOA). Poursuivant l'une des idées proposées, FOA après l'avoir fusionné avec PSO nous a permis d'optimiser les positions initiales des nœuds inconnus estimées par la célèbre technique de localisation Centroid.

Un troisième aspect du travail concerne la faisabilité d'une approche qui repose sur une méta-inférence FOA améliorée. Des simulations dans la plate-forme Matlab et des comparaisons avec d'autres méta-inférences robustes de localisation nous ont donné un aperçu des performances de notre proposition.

Pour montrer la performance de nos contributions, une étude empirique a été réalisée ainsi que les résultats de simulation ont été comparés aux résultats de l'algorithme CSO amélioré.

Diverses évaluations effectuées ont montré que l'algorithme FOA amélioré est significativement affecté par les paramètres du réseau et de la méthode, et nous a également permis de surveiller sa supériorité par rapport au CSO amélioré en termes de précision de localisation. Pour les travaux futurs, nous conseillons aux chercheurs de proposer des versions parallèles de ce type d'algorithme afin de gagner du temps d'exécution et d'application dans un environnement 3D et dans des réseaux de capteurs mobiles.

# **Bibliographie**

- [1]: N. Siddique and H. Adeli, "Nature Inspired Computing: An Overview and Some Future Directions," Cognitive Computation, vol. 7, no. 6, pp. 706–714, Nov. 2015
- [2] Y. Challal, Réseaux de Capteurs Sans Fils, Support de cour, 2008

Yaser Youcef, Routage pour la gestion de l'énergie dans les réseaux de capteurs sans fil.

- [3] BELATTAR, KARIMA. Réalisation d'une Application Mobile-Health avec un réseau corporel sans fil pour le suivi d'une personne âgée. Diss. FACULTE DES MATHEMATIQUES ET DE L'INFORMATIQUE DEPARTEMENT D'INFORMATIQUE-Spécialité: Systèmes d'Informations Avancés/Réseaux et Technologie de l'Information et de Communication. 2020.
- [4] https://www.cours-gratuit.com/cours-arduino/tutoriel-arduino-station-meteo-projet
- [5] Computer Science. Université de Haute Alsace Mulhouse, 2010

David Martins, Sécurité dans les réseaux de capteurs sans fil Stéganographie et réseaux

- [6] de confiance, Thèse de doctorat, Université de Franche-Comté, 2010
- [7]. Dargie, Waltenegus, and Christian Poellabauer. Fundamentals of wireless sensor networks: theory and practice. John Wiley& Sons, 2010.
- [8] CHALLAL. Yacine. Réseaux de capteurs sans fils : Support cours. Ecole nationale Supérieure d'Informatique, Algérie. Novemdre 2008 https://moodle.utc.fr/file.php/498/support-SIT60.pdf
- [9]: Ammar, Amira Ben,<< Approche cross-layer pour la minimisation d'énergie dans les réseaux de capteurs sans fil>>,Submitted on 18 Jan 2019
- [10]: BOURAI AMAR et BENTABET Abdel Hamid, << Mise en œuvre d'un réseau de senseurs sans fil pour la détection et la localisation d'un incendie>>, la promotion LMD Master RMS.
- [11]: KHALILI Zeyneb et BOUCHRA Meryem, << Approche distribuée de localisation des nœuds dans les réseaux de capteurs sans fil>>, Mémoire de fin d'étude, en vue de l'obtention du diplôme de licence en informatique, Université Ahmed Draia Adrar ,2016/2017.
- [12]: CHETOUAH Karima et BOUALLALA Khadidja, << Auto-localisation des nœuds dans les réseaux de capteur sans fil par la méthode Multi-DimensionalScaling>>, Mémoire de fin d'étude, en vue de l'obtention du diplôme de licence en informatique option :-Ingénierie des systèmes d'informatique et logiciels (ISIL)-Système d'information (SI), 2016/2017.
- [13]: Van der Haegen Mathieu, << Réseaux de senseurs sans fil : problèmes de localisation>>, Mémoire proposé en vue de l'obtention du grade de licencié en informatique, Bruxelles, 2007.

- [14]: MELLAOUI Mohamed et OUARAB Tamazight, << Localisation d'un Mobile par réseaux de capteurs sans fil>>, Mémoire en vue de l'obtention du déplome de mastar domaine : sciences et technologies filière : télécommunications spécialité : système des télécommunications, 21 juin 2018.
- [15]: Mr.BELDJOUDIFawzi et Mr.BOUKHANOUF Imad,<< ÉTUDE DES ALGORITHMES DE LOCALISATION DANS LES RÉSEAUX DE CAPTEURS SANS FIL>>, Mémoire de fin de cycle en vue de l'obtention du diplôme de master professionnel en informatique option : Administration et Sécurité des Réseaux, 2017/2018.
- [16]G. Bontempi. Statistical Foundations of Machine Learning. Universite Libre de Bruxelles, 2006
- [17] Kara, Messaoud. Réseau de capteurs sans fil : étude en vue de la réalisation d'un récepteur GPS différentiel à faible coût. Diss. Université Blaise Pascal-Clermont-Ferrand II, 2009
- [18]: Y. Shang et W. Ruml, et Y. Zhang et M. Fromherz, << Localization from connectivity in sensor networks>>, IEEE Transactions on Parallel and Distributed Systems, 15(11), (2004), 961-974.
- [19]: C. Savarese et J. Rabaey, <<Robust positioning algorithms for distributed ad-hoc wireless sensor networks>>, Proceedings of the General Track: 2002 USENIX Annual Technical Conference, (2002), (pp. 317-327).
- [20]: DALCE Réjane, << Méthodes de localisation par le signal de communication dans les réseaux de capteurs sans fil en intérieur>>, Institut National des Sciences Appliquées de Toulouse (INSA de Toulouse),26 juin 2013.
- [21] Bulusu, Nirupama, John Heidemann, and Deborah Estrin. "GPS-less low-cost outdoor localization for very small devices." IEEE personal communications 7.5 (2000): 28-34.
- [22]: XianbingMeng, Yu Liu, Xiaozhi Gao, Hengzhen Zhang, "A new bio inspired algorithm: Chicken Swarm Optimization," Advance in swarm intelligence, Vol. 8794, pp. 86-94, 2014.
- [23] CHALANE Idir. et OUARI Tarek. « Optimisation des paramètres d'un PID par essaims particulaires(PSO). » mémoire de Master Université A. MIRA de Bejaia 2017
- [24] BENDAHOUA Sarah « rapport sur le recuit simulé » Université d'Oran Mohamed Boudiaf 2012/2013
- [25]: Kennedy, J., Eberhart, R.: Particle swarm optimization. In: IEEE Intenationl Conference on Neural Networks, vol. 4, pp. 1942-1948(1995).
- [26]: RABHI Seddik. "Optimisation des algorithmes de localisation dans les réseaux de capteurs sans fil", informatique, DOCTORAT 10/04/2021, Université Ahmed Draia Adrar.
- [27]: Bulusu, N., Heidemann, J., Estrin, D.: GPS-less low-cost outdoor localization for very small devices. IEEE Personal Communications, vol. 7, no. 5, Oct. 2000, pp. 28-34

[28]: Melle Djiral Amina,<<Implémentation et réalisation d'un algorithme de localisation « Centroid » dans les RCSF>>,Mémoire de fin d'études Pour l'obtention du diplôme de Master en Informatique ,Université de Telemcen ,2015.

[29]: Jingang Cao « A Localization Algorithm Based on Particle Swarm Optimization and Quasi-Newton Algorithm for Wireless Sensor Networks » Department of Computer, North China Electric Power University, Baoding 071003, China

[30]: CHETOUAH Karima &BADDOU Saliha. "Application de l'essaim de poulet pour la localisation dans les RCSFs", Système intelligents, Mémoire de fin d'étude, en vue de l'obtention du diplôme de Masteren informatique 2018/2019, Université Ahmed Draia - Adrar.