

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique
Université Alhadj Lakhder de Batna
Faculté des Sciences



Mémoire



Présenté Par

Samiha REBIAI

En vue de l'obtention du diplôme de Magister

Filière : informatique

Option : Systèmes informatiques Intelligents et
Communicants

Titre

***Routage Adaptatif dans les Réseaux de
Robots Mobiles Autonomes***

Soutenue le : 03 Mars 2015

Devant le jury composé de :

Président **A. LAHLOUHI**

MCA., Université de Batna

Rapporteur **K.E. MELKEMI**

MCA., Université de Biskra

Examineurs **S.ZIDAT**

MCA., Université de Batna

A.BACHIR

MCA., Université de Biskra

Invité **M.R.ABDESSEMED**

MCB., Université de Batna

Remerciements

Je tiens tout d'abord à remercier *ALLAH* le tout Puissant et Miséricordieux, qui nous a donné la force et la patience pour accomplir ce travail.

EL HAMDOULILLAH

Je tiens à remercier *Docteur Kamel Eddine MELKEMI* mon encadreur pour avoir bien voulu diriger mon mémoire et qui a su me conseiller. Je voudrais également le remercier pour ses conseils et ses encouragements.

Je tiens à remercier *Docteur Mohamed-Rida ABDESSEMED* pour sa disponibilité, ses idées et ses encouragements.

Mes vifs remerciements vont également aux *membres du jury* pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner ce travail et de l'enrichir par leurs propositions.

Mes remerciements vont également à tous mes enseignants du département d'informatique.

Je souhaite à tous mes *collègues de magistère* une bonne continuation vers la réussite.

Merci à tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail.

Dédicaces

À mes chers parents,

À mes frères et sœurs

À Amoula

À ma famille

À tous mes enseignants,

*Aux vrais amis qui sont toujours présents, même
dans les moments difficiles.*

Je dédie ce travail

Résumé

D'une part, la mobilité des nœuds représente une caractéristique clé dans les réseaux mobiles ad hoc (MANETs), d'autre part le routage dans ces mêmes réseaux est basé sur l'existence d'une connectivité debout-en-bout. Une forte mobilité entraîne souvent des problèmes de déconnexion. Restreindre la mobilité des nœuds pour le maintien de la connectivité n'est pas une solution valable dans tous les scénarios. L'exploration est un exemple d'application qui oblige les nœuds (robots ou capteurs) à s'éloigner les uns des autres autant que possible afin d'explorer rapidement l'ensemble de la zone ciblée. Mais, d'autre part, chaque nœud doit conserver la connectivité avec les membres de son équipe afin de pouvoir communiquer. A cet effet, les protocoles proposés pour les réseaux tolérant au délai (Delay Tolerant Networking (DTN)) sont considérés comme les plus appropriés à utiliser dans les réseaux avec des déconnexions fréquentes. Cela nous a motivé à proposer un protocole adaptatif permettant de basculer entre des protocoles orientés MANET et d'autres orientés DTN, guidé par l'état de connectivité du réseau ciblé : a) un MANET si les robots ne constituent pas les nœuds du réseau, ou b) un RANET (Robot Adhoc Network ou RANET) sinon. Cela Pour atteindre une qualité d'exploration aussi élevée que possibles. Un phénomène de routage adaptatif émerge au macro-niveau du système RANET permettant d'observer comment un réseau de robots arrive à surmonter le problème de déconnexion pour réussir une tâche d'exploration.

Mots clés : Connectivité, Mobilité, Communication, MANET, DTN, Routage adaptatif, RANET.

Abstract

Mobility is a key characteristic of nodes in Mobile Ad hoc Network (MANET) but at the same time MANET is based on the existing of end-to-end connectivity. The high mobility often resulting of connectivity problems. Restrict node's mobility to maintaining connectivity is not available solution in all scenarios. Exploration is an application example, which needs nodes (robots or sensors) to move away from each other as far as possible to explore quickly the whole target area. But, on the other hand each node must keep connectivity with its team members in order to communicate. At the same time, Delay Tolerant Networking (DTN) protocols are considered as the most suitable to use in networks with frequent disconnections. This, motivate us to suggest an adaptive protocol idea, allowing to switch between the MANET and DTN protocols, according to the connectivity situation of a particular MANET (Robot Adhoc Network or RANET). Thus, to achieve an exploration task with best possible performances. An adaptive routing phenomenon emerges at the high level of the RANET leading to overcome disconnection problems in cooperative robot exploration task.

Index Terms—connectivity, mobility, communication, MANET, DTN, adaptive routing, RANET.

ملخص

تعد الحركة من الصفات الأساسية لعقد شبكة MANET. لكن في نفس الوقت هذا النوع من الشبكات يعتمد على وجود اتصال بين المرسل و المستقبل. الحركة الكثيرة تتسبب في مشكل انقطاع الاتصال. تقييد حركة العقد لضمان الاتصال ليسا حلا ناجحاً دائماً. عملية الاستكشاف من المهام التي تستوجب تباعد العقد (روبوتات أو أجهزة استشعار) لاستكشاف المنطقة المقصودة سريعاً. لكن في نفس الوقت، يستوجب على العقد البقاء قريبين من بعضهم للحفاظ على الاتصال بينهم. في حالة الشبكات ذات الانقطاع المتكرر، تعد بروتوكولات DTN الأنسب للتواصل. هذا السبب دفع بنا لاقتراح فكرة بروتوكول متكيف يسمح بالتبديل بين بروتوكولات شبكة MANET و بين بروتوكولات شبكة DTN وفقاً لحالة الاتصال لشبكة MANET خاصة (شبكة الروبوتات اللاسلكية: RANET). هذا كله لتحقيق مهمة الاستكشاف مع أفضل أداء ممكن. ظاهرة التوجيه التكييفي يظهر على مستوى عال من RANET للتغلب على مشكلة انقطاع الاتصال في عملية استكشاف من طرف الروبوتات المتعاونة.

الكلمات الرئيسية: الربط، التنقل، الاتصال، DTN، MANET، التوجيه المتكيف، RANET.

Liste des acronymes

MANET	M obile A dhoc NET work
DTN	D elay T olerant N etwork
POI	P oint O f I nterest
TTPS	T aux de T ransmission de P aquets avec S uccès
TCPI	T aux de C ouverture des P oints d' I ntérêt
GA	G enetic A lgorithm
PSO	P artice S warm O ptimisation
ACO	A nt C olony O ptimization

Table des matières

Liste des Acronymes	I
Tables des matières	II
Liste des figures	VI
Liste des tableaux	IX
Liste des équations	X
Introduction générale	1

Chapitre 1: Robotique de groupe

1.1.Introduction.....	4
1.2.Intelligence en essaim	4
1.3.Robotique	6
1.3.1. Introduction	6
1.3.2. Architecture générale d'un robot	7
1.3.3. Architecture de contrôle d'un robot mobile autonome	8
1.3.3.1. Architecture de contrôle cognitive.....	9
1.3.3.2. Architecture de contrôle réactive	9
1.3.3.3. Architecture de contrôle centralisée	9
1.3.3.4. Architecture de contrôle décentralisée	9
1.3.4. Domaines d'application	9
1.3.5. Réseaux de robots	10
1.3.5.1. Définition	10
1.3.5.2. Communication	11
1.3.5.3. Coopération, coordination et collaboration	14
1.3.5.4. Domaines d'applications.....	15
1.4.Conclusion	17

Chapitre 2: Réseaux sans fil

2.1.Introduction.....	19
2.2.Les réseaux sans fil.....	19
2.2.1.Classification des réseaux sans fil.....	20
2.2.2. Réseau Ad hoc	21
2.2.2.1. Introduction.....	21
2.2.2.2. Piles protocolaires	22
2.2.2.3. Routage dans les réseaux Ad hoc.....	24
2.2.2.4. Problèmes du routage dans les réseaux Manet.....	24
2.2.2.5. Les applications des réseaux ad hoc	30

Table des matières

2.2.3. Réseaux DTN.....	30
2.2.3.1. Introduction	30
2.2.3.2. Définition des réseaux DTN.....	31
2.2.3.3. Caractéristiques des réseaux DTN.....	32
2.2.3.4. Fonctionnement des réseaux DTN.....	32
2.2.3.5. Routage dans les réseaux DTN	34
2.2.3.6. Domaines d'application de réseaux DTN	42
2.2.4. Le réseau Ad hoc et le réseau DTN	43
2.3. Conclusion	44

Chapitre 3: Méthode d'optimisation (Méta-heuristique PSO)

3.1. Introduction	46
3.2. Classification des méthodes d'optimisation.....	46
3.2.1. Méthodes exactes (déterministes)	47
3.2.2. Méthodes approchés (non déterministes).....	48
3.2.2.1. Les heuristiques.....	48
3.2.2.2. Les méta-heuristiques.....	48
3.2.2.2.1. Les algorithmes génétiques.....	49
3.2.2.2.2. La programmation génétique.....	49
3.2.2.2.3. Optimisation par colonies de fourmis.....	49
3.2.2.2.4. Optimisation par essaim particulaire	49
3.3. Avantages et inconvénients des méthodes approchés.....	53
3.4. Conclusion.....	53

Chapitre 4: Tache d'exploration et de couverture

4.1. Introduction	56
4.2. Problème d'exploration.....	56
4.2.1. Introduction	56
4.2.2. Localisation et cartographie	57
4.2.3. Problèmes liés à la tâche d'exploration.....	59
4.2.3.1. Maintien de la connectivité.....	59
4.2.3.2. Minimisation du temps d'exploration	60
4.2.3.3. Minimisation d'énergie	61
4.2.4. Travaux sur le problème d'exploration	61
4.3. Couverture	63
4.3.1. Zone de couverture	63
4.3.2. Taxonomie de types de couverture	63
4.3.2.1. Surveillance de l'environnement.....	63
4.3.2.1.1. Couverture de zone.....	63

Table des matières

4.3.2.1.2. Couverture de points.....	64
4.3.2.1.3. Couverture de frontière.....	64
4.3.2.2. Types de couverture	64
4.3.2.2.1. Couverture simple.....	64
4.3.2.2.2. Couverture multiple.....	64
4.3.2.3. Dynamicité de la couverture	65
4.3.2.3.1. Couverture statique	65
4.3.2.3.2. Couverture dynamique	65
4.4. Conclusion.....	65

Chapitre 5: Problème posé et solution proposée

5.1. Introduction	67
5.2. Problème posé	67
5.3. Analyse du problème.....	68
5.3.1. Coopération	68
5.3.2. Conditions de communication et paramètres à régler	68
5.3.3. Fonction objective	69
5.3.4. Exemples	70
5.3.5. Communication horizontale.....	70
5.3.6. Communication verticale	70
5.4. Solution	71
5.4.1. Ressources logiques utilisées.....	71
5.4.2. Ressources physiques (Modélisation du robot).....	72
5.4.3. Paramètre de La coopération	72
5.4.4. Paramètre de l'adaptation.....	75
5.5. Particle Swarm Optimization	77
5.5.1. Principe	77
5.5.2. Valeurs des paramètres du PSO utilisés	78
5.5.3. Algorithme de PSO	78
5.6. Algorithme d'exploration	79
5.6.1. Création de la population	79
5.6.2. Calcul d'utilité	80
5.7. Protocoles de routage	81
5.7.1. MANET	81
5.7.2. DTN.....	82
5.8. Algorithme de simulation	83
5.9. Conclusion	84

Chapitre 6: Mise-en-œuvre et résultats de simulation

6.1. Introduction	
6.2. Mise en œuvre de la simulation	86
6.2.1. Introduction	86

Table des matières

6.2.2. Choix de simulateur	86
6.2.3. Mise en œuvre de la solution proposée	87
6.3. Résultats de simulation	89
6.3.1. Les règles décelées	92
6.3.2. La phase de la mise à l'échelle	93
6.3.2.1. La mise à l'échelle du nombre de robots	93
6.3.2.2. La mise à l'échelle du nombre de points d'intérêt.....	94
6.3.2.3. La mise à l'échelle de la taille d'environnement	94
6.3.2.4. La mise à l'échelle du rayon de détection.....	94
6.3.3. La mise à l'échelle du rayon de communication	96
6.4. Comparaison des résultats	97
6.5. Conclusion	100
Conclusion générale.....	101
Références bibliographiques.....	103

Liste des figures

Fig1.1. Essais naturels et artificiels	5
Fig1.2. Cinq mécanismes pour un comportement de groupe.....	5
Fig1.3. Communication d'un réseau de robots en essaim	6
Fig1.4. Les premiers robots développés	6
Fig1.5. Interactions d'un robot avec son environnement	7
Fig1.6. Architecture générale d'un robot mobile	8
Fig1.7. Classification d'architecture de contrôle.....	8
Fig1.8. Domaines d'application de robotique.....	10
Fig1.9. Réseau téléopéré.....	11
Fig1.10. Réseau de robots autonome	11
Fig1.11. Évolution des technologies de communication de robots sans fil.....	12
Fig1.12. Classification de la communication selon le mode d'interaction	12
Fig1.13. Classification des types de communication	13
Fig1.14. Petits modules.....	15
Fig1.15. Déploiement des robots dans un environnement inconnu pour détecter les intrus	15
Fig2.1. Classification des réseaux sans fils	20
Fig2.2. Modèle d'un réseau mobile avec infrastructure	21
Fig2.3. Modèle d'un réseau mobile sans infrastructure	21
Fig2.4. Réseau Ad hoc	22
Fig2.5. Représentation de la structure en couches des réseaux de robots mobiles	24
Fig2.6. Classification des protocoles de routage dans un réseau Ad hoc.....	25
Fig2.7. Le routage dans le protocole AODV	26
Fig2.8. Le routage dans le protocole OLSR	27
Fig2.9. Le protocole hybride ZRP.....	28
Fig2.10. Topologie d'un réseau de clusters	29
Fig2.11. Topologie d'un réseau de BACKBONE	29
Fig2.12. Exemple d'un réseau DTN	31
Fig2.13. Principe de fonctionnement d'un réseau DTN	33
Fig2.14. La technique Store and Forward.....	33
Fig2.15. Comparaison entre les couches Internet et les couches DTN	34
Fig2.16. Diagramme résumant la classification de Jain & Al [65].....	36
Fig2.17. Extension du diagramme de classification de Jain & Al [66].....	36
Fig2.18. Classification des protocoles de routage DTN selon le taux de consommation de ressources.....	37
Fig2.19. Classification de Gong et al [63]	38
Fig2.20. Classification de Min et al [6]	39
Fig3.1. Classification des méthodes d'optimisation.....	47
Fig3.2. Pseudo code des algorithmes évolutionnistes.....	48

Liste des figures

Fig3.3. Principe de l'AG	49
Fig3.4. Schéma de principe de recherche de PSO.....	50
Fig3.5. Principe de PSO.....	51
Fig4.1. Les trois parties de la tâche d'exploration	56
Fig4.2. Grille d'occupation d'un couloir de laboratoire.....	58
Fig4.3. Représentation topologique	58
Fig4.4. Aperçu de la carte locale utilisée pour l'exploration	62
Fig4.5. Méthode de triangulation dynamique	62
Fig4.6. Portées d'un robot.....	63
Fig4.7. Couverture de Zone	63
Fig4.8. Couverture des points d'intérêt	64
Fig4.9. Couverture de frontière.....	64
Fig5.1. L'environnement d'exploration.....	68
Fig5.2. Diagramme de transition d'états	74
Fig5.3. Solution via PSO	77
Fig5.4. Les différentes directions considérées d'un robot dans notre algorithme d'exploration	79
Fig6.1. Processus de simulation.....	86
Fig6.2. La vision locale de robot.....	87
Fig6.3. L'occupation des cellules qui entourent le robot	87
Fig6.4. Structure de la table de consultation coopérative.....	88
Fig6.5. Structure de la table de consultation concernant l'adaptation	89
Fig6.6. Diffusion du message par un robot bloqué	90
Fig6.7. Diffusion de message lors de détection des POI non couverts.....	90
Fig6.8. Valeurs de Gbest courant et Gbest dans les cinquante itérations.....	91
Fig6.9. Valeurs de TCPI courant et TCPI dans les cinquante itérations	91
Fig6.10. Valeurs de TTPS courant et TTPS dans les cinquante itérations	92
Fig6.11. Mécanisme d'adaptation	93
Fig6.12. Mise à l'échelle du nombre de robots	93
Fig6.13. Mise à l'échelle du nombre de POI	94
Fig6.14. Mise à l'échelle de la taille d'environnement	94
Fig6.15. Mise à l'échelle du rayon de détection	95
Fig6.16. Le taux de couverture des points d'intérêt en fonction de la taille d'environnement et le rayon de détection	95
Fig6.17. Mise à l'échelle du rayon de communication	96
Fig6.18. Le taux de transmission de paquets avec succès en fonction du rayon de détection.....	96
Fig6.19. Comparaison entre EAMC et EARA par rapport au temps d'exploration en fonction de nombre de robots	98
Fig6.20. Comparaison entre ESMC et EARA par rapport au temps d'exploration en fonction du nombre de robots	98
Fig6.21. Comparaison entre ESMC, EAMC et EARA par rapport au temps d'exploration en fonction de nombre de robots	99

Liste des figures

Fig6.22. Comparaison entre ESMC et EARA par rapport au TCPI en fonction de taille d'environnement	99
Fig6.23. Comparaison entre ESMC et EARA par rapport au TTPS en fonction de nombre de robots.....	100

Liste des tableaux

Tab1.1	Tableau comparatif entre les technologies de communication des robots	12
Tab2.1	Comparaison entre les différents types de réseaux sans fil	20
Tab2.2	Représentation de la structure en couches des réseaux mobiles Ad hoc par rapport à OSI et TCP/IP	23
Tab3.1	Applications des algorithmes de PSO	53
Tab6.1	Paramètres de l'algorithme de PSO	90
Tab6.2	Paramètres de simulation	90

Liste des équations

Equation1 : Prédicibilité de livraison	41
Equation2 : Mise à jour de prédictibilité de livraison	42
Equation3 : Influence de transitivity sur la prévisibilité de livraison.....	42
Equation4 : Mise à jour de vitesse de la particule	50
Equation5 : Mise à jour de position de la particule	50
Equation6 : Fonction objective	69
Equation7 : Force d'attraction de message.....	70
Equation8 : Taux local d'échec	76

Introduction générale

Un réseau de robots mobile est composé d'une collection distribuée de robots. Chacun de ces robots ayant des capacités de détection, de calcul, de communication et de mobilité employés pour accomplir des tâches coopératives telles que l'exploration, la localisation et le routage de messages.

L'autonomie, la capacité d'interaction avec les personnes et l'environnement et les facultés d'apprentissage représentent les aspects individuels du robot [1] qui n'est pas toujours aussi efficace que l'on peut espérer. Bon nombre de tâches complexes restent hors de sa portée et même s'il arrive à venir à bout de quelques-unes de ses missions, beaucoup de problèmes subsistent. Un exemple type de ceci est la situation où il tombe en panne, ce qui met définitivement fin à sa mission. De ce fait, la réalisation de ce genre de tâches, difficiles, voir même impossibles à accomplir par un seul robot, nécessite le recours à une collectivité de robots, exigeant souvent la coopération entre ses différents membres [2][3]. Évoquer la coopération nous amène à parler de la communication où l'échange peut se faire de manière directe ou indirecte.

Dans ce travail nous focalisons sur les échanges directs, basés sur la mise en place d'un réseau informatique. Ceci nous fait transiter vers un autre domaine qui commence à prendre une envergure de plus en plus importante dans le domaine de la recherche et à présenter un besoin accru au niveau de certains secteurs aussi importants que la défense militaire, l'industrie, l'agriculture, la santé et la sécurité. Il s'agit du domaine des réseaux de robots, qui est lié étroitement à celui de la robotique collective. Dans ces secteurs, des tâches comme l'exploration de zones à haut risque ou bien inaccessibles et la recherche de survivants dans un milieu hostile sont des applications potentielles pour les réseaux de robots mobiles autonomes et auto-organisant.

L'exploration de l'environnement est l'un des problèmes les plus étudiés en robotique collective [3][4]. Le problème c'est que pour que les robots puissent couvrir la plus grande superficie possible de l'environnement en un temps optimal, ils doivent s'éloigner, simultanément, les uns des autres autant que possible. Ceci élimine l'hypothèse du contrôle des robots en vue de les maintenir en formation serrée, étant donné que pour communiquer ils doivent rester dans la portée des uns les autres ; çà leur permet de préserver la connectivité entre eux.

Pour répondre à ce problème de connectivité, nous proposons une solution où un routage à deux niveaux est utilisé pour essayer de s'adapter à l'état du réseau. Notons que l'état du réseau est exprimé par les positions de ses nœuds (robots dans notre cas) dictée par l'algorithme d'exploration adoptée.

Le routage dans les réseaux mobiles est souvent peu adaptatif. Il suppose soit une connectivité disponible debout en bout (comme le routage réactif ou proactif dans les réseaux MANET [5]) soit une connectivité complètement intermittente (comme le routage DTN [6]). Un routage adaptatif, par contre, doit s'accommoder aux situations de la disponibilité de la connectivité et de l'intermittence de cette dernière pour livrer

Introduction générale

le paquet de données avec succès dans toutes ces situations tout en garantissant la réalisation de la tâche ciblée.

Pour atteindre cet objectif, une méthode d'optimisation inspirée du comportement social de certain type d'animaux et d'insectes est utilisée. Elle est devenue en peu de temps l'une des méthodes les plus efficaces dans le domaine de l'optimisation analytique permettant de réduire le temps de traitement, l'occupation de la mémoire et d'améliorer la qualité des solutions. Il s'agit d'utiliser un ensemble de règles de contrôle locales, assez simples, de plusieurs essaims biologiques (comme les gènes, les fourmis, les abeilles ou les oiseaux) pour proposer de nouvelles approches de recherche intelligentes de la solution optimale telles que les algorithmes génétiques (GA) [7], l'optimisation par essaim particulaire (PSO) [8], les algorithmes de fourmis (ACO)[9]. Notons que la coordination entre les robots en essaim a été l'un des domaines qui a bénéficié le plus de l'inspiration biologique en générale et du PSO en particulier. C'est pour cette raison que nous avons décidé d'appliquer l'approche PSO pour résoudre le problème décrit ci-dessus.

Pour ce faire, ce mémoire est organisé en six chapitres.

Dans le premier chapitre nous survolons le domaine de la robotique en groupe en focalisant tout particulièrement sur la robotique collective et les réseaux de robots.

Le deuxième chapitre est consacré aux réseaux sans fil où nous mettons l'accent, dans sa première partie, sur les réseaux Manet et, dans sa deuxième partie, sur les réseaux DTN.

Dans le troisième chapitre nous présentons les différentes méthodes d'optimisation en insistant sur la méthode d'Optimisation par essaim particulaire(PSO).

Dans le quatrième chapitre nous décrivons le problème d'exploration et de couverture comme nous parlons de la localisation, de la communication et du problème de connectivité. Aussi, nous passons en revue les différentes solutions existantes.

Dans le cinquième chapitre nous exposons le problème traité dans ce projet et la solution proposée.

Dans le sixième et dernier chapitre nous expliquons les détails de la mise-en-œuvre et nous présentons et interprétons les résultats de simulation (les règles à utiliser pour un routage adaptatif) obtenus via NetLogo. Pour montrer la consistance de ces règles, nous les soumettons à une phase de mise à l'échelle ainsi qu'à une comparaison avec un protocole de référence.

Nous clôturons ce mémoire par une conclusion où on fait une récapitulation succincte du travail ainsi accompli et où nous proposons quelques perspectives à prendre en considération de nos futur travaux.

1

CHAPITRE

Robotique de groupe

Chapitre 1 : Robotique de groupe

1.1.Introduction

L'auto-organisation est un phénomène émergent résultant d'une multitude d'interactions entre différents membres d'une même société qu'elle soit naturelle (comme les insectes sociaux) ou artificielle (comme les collectivités de robots). Elle peut aussi résulter d'une interaction entre la société, en question, et l'environnement. L'absence de contrôle hiérarchique ou centralisé permettant de gérer les activités entre les insectes sociaux incitent à considérer ces derniers comme un modèle pouvant être appliqué dans le domaine de la robotique. Il est clair que l'intelligence artificielle est inspirée du comportement individuel de l'être humain ; elle cherche à reproduire son raisonnement. Cependant, la nature abonde de formes d'intelligence collective. La nouveauté dans ce contexte c'est qu'au lieu de prendre en compte une intelligence individuelle on tient compte de celle de la collectivité.

L'intelligence en essaim est une forme particulière de ce type d'intelligence ; elle se base sur un essaim d'individus pourvus d'une intelligence minimale, mais totalement autonomes. La robotique en essaim est actuellement l'un des plus importants domaines d'application de l'intelligence en essaims, elle est une branche de la robotique collective axée sur l'étude des groupes relativement importants de robots avec des capacités de communication et de détection limitées [10]. Un réseau de robots est composé d'un ensemble de machines intelligentes mobiles homogènes ou hétérogènes en interaction dans le but de réaliser des tâches difficiles voire même impossibles à accomplir par un seul robot. L'utilisation d'un réseau de robots présente plusieurs avantages en comparaison avec un robot solitaire (ou unique) tel que le parallélisme et la robustesse (la souplesse, l'efficacité, le gain économique, ...). Le réseau de robots est utile dans plusieurs domaines d'applications. Comme celles : militaires, de surveillance, de transport, d'exploration spatiale ou sous-marine [11].

Ce chapitre s'intéresse à la robotique de groupe en générale, mais focalise sur la robotique collective et plus particulièrement sur celle en essaim. Pour se faire il est nécessaire d'introduire, en premier lieu, aussi bien le concept de l'intelligence en essaim (cas particulier de l'intelligence collective) que celui de la robotique. Il serait, par la suite, plus aisé d'expliquer les réseaux de robots.

1.2.Intelligence en essaim

Les chercheurs ont pu transformer des modèles du comportement collectif des insectes sociaux en méthodes utiles pour résoudre des problèmes d'optimisation. L'intelligence en essaim est un nouveau domaine de recherche quia vu le jour dans les dernières décennies et dont l'objectif est de transformer la connaissance que les éthologues ont des capacités collectives que les insectes sociaux exhibent pour venir à bout de leurs problèmes en techniques artificielles permettant de trouver des solutions à des problèmes humains [12]. Le terme «*essaim*» fait référence à un grand groupe d'individus qui interagissent localement avec des objectifs communs. Il est utilisé pour décrire tous les types de comportements collectifs. L'intelligence en essaim est l'intelligence collective qui émerge à partir des interactions entre des groupes

Chapitre 1 : Robotique de groupe

d'individus autonomes de grandes tailles [13]. La nature nous montre des exemples typiques, comme celles rencontrés dans les espèces d'insectes sociaux. En effet, des colonies de fourmis, des guêpes, des abeilles et d'autres types d'insectes vivant en société ont l'intuition d'avoir des comportements complexes sans l'intervention d'un superviseur et sans aucun contrôle central dictant la manière de réalisation de tels comportements. Les animaux exécutent des comportements simples pour permettre l'émergence de comportements complexes qui sont essentiels à l'exploration de nouveau territoire, la recherche de nourriture, la chasse, la construction des nids et la survie [14].



Figure 1.1 Essaims naturels et artificiels : l'ordre est de gauche à droite et du haut vers le bas : (a) : robotique en essaim, (b) et (e) : colonie de fourmis cherche de nourriture, (c) : nuée d'oiseaux, (d) : Banc de poisson et la dernière figure (f) montre l'application du principe d'une colonie de fourmis dans le problème du voyageur de commerce pour obtenir le plus court chemin.

La robotique collective est synonyme de l'existence d'un groupe de robots. Elle stipule non seulement la mise-en-place d'un contrôle individuel pour chaque robot, mais impose aussi l'utilisation de stratégies de contrôle appropriées afin que l'assemblage de toutes ces entités robotiques engendre des configurations cohérentes et efficaces pour la réalisation des tâches désirées [15]. Kube et Zhang [16] observent les communautés d'insectes et proposent un mécanisme de contrôle en temps réel basé sur cinq mécanismes, reposant sur l'idée de promouvoir la collaboration plutôt que l'effort individuel (voir figure 1.2).

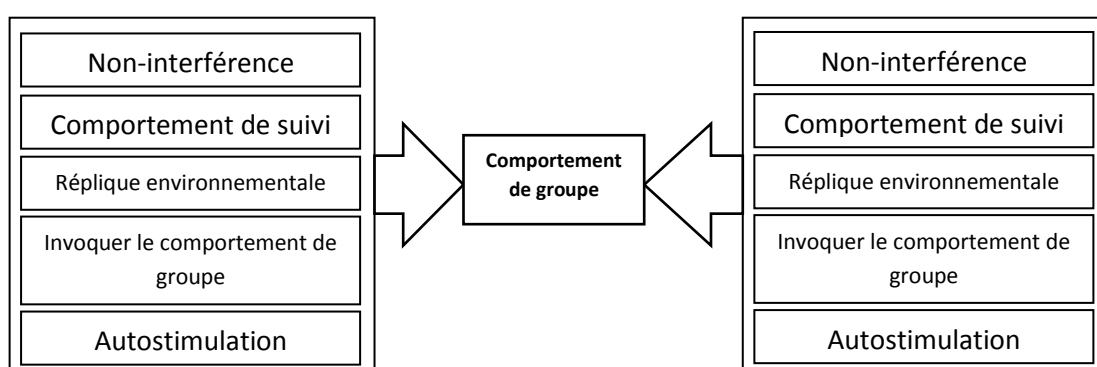


Figure 1.2 : Cinq mécanismes pour un comportement de groupe

Le premier mécanisme « *la non-interférence* » est une forme simple de coopération où les robots n'interfèrent pas avec les actions des uns les autres. Le comportement de

Chapitre 1 : Robotique de groupe

robot-avoidance (ou évitement des autres robots) est un exemple implémenté par [16] comme mécanisme de non-interférence. *Le comportement de suivi* est un mécanisme qui rassemble les robots en groupe pour accomplir leur tâche commune. Le troisième mécanisme utilise une *réplique environnementale* pour invoquer le comportement de groupe. Chaque robot doit être capable *d'invoquer le comportement de groupe* c'est-à-dire il peut reconnaître s'il se trouve dans un groupe grâce aux senseurs appropriés. Le dernier mécanisme «l'autostimulation» permet d'invoquer le comportement de groupe. Par exemple un robot, ayant trouvé une tâche à effectuer, va demander l'aide des membres de son groupe pour accomplir sa tâche. La technique d'intelligence en essaim est largement utilisée dans les tâches robotiques tel que le déplacement autour des obstacles, pousser un bloc, etc. Un exemple d'architecture proposée d'un réseau de communication de robots en essaim est illustré dans la figure 1.3. L'intelligence en essaim utilise des agents logiciels mobiles pour la gestion du réseau. Ces agents sont des entités autonomes, à la fois proactives et réactives et ils ont la capacité de s'adapter, de coopérer et de se déplacer intelligemment d'un endroit à un autre dans le réseau [17].

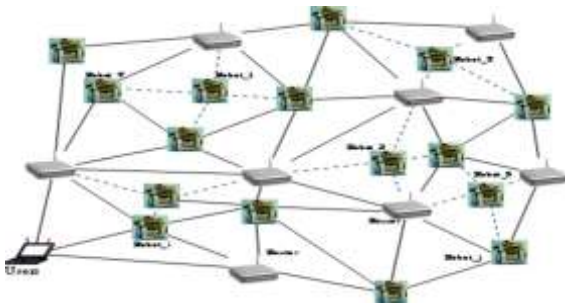


Figure 1.3 Communication d'un réseau de robot: Les routeurs sans fil maillés sont déployés dans l'environnement, et chacun d'eux se compose de plusieurs antennes qui l'amènent à fonctionner sur de multiples canaux pour améliorer la capacité du réseau et de couverture. Chaque robot est équipé par des adaptateurs sans fil qui lui permet de communiquer avec les routeurs et les autres robots du même réseau.

1.3. Robotique

1.3.1. Introduction

Le terme « robot » est introduit par Karel Capek, auteur Tchèque, dans sa pièce Russel's Universal Robots, en 1921. Ce terme vient des mots tchèques « robota » qui signifie (travail) et « robotnik » qui indique (ouvrier) [18].

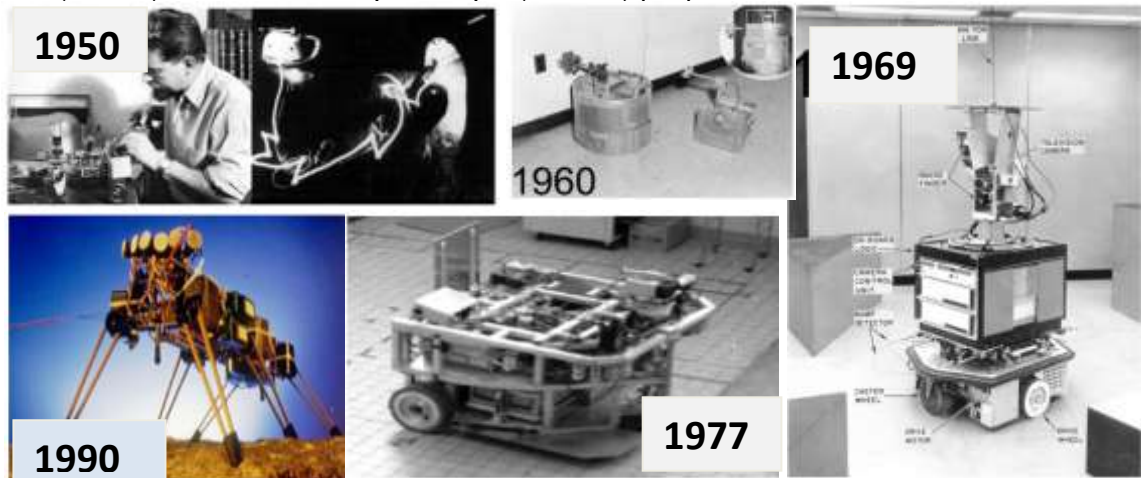


Figure 1.4 les premiers robots développés : cette figure montre la Tortue construite par Grey Walter dans les années 1950 (nommée "machina speculatrix" et surnommée Elsie), le robot "Beast" de l'université John Hopkins dans les années 1960, le robot Shakey de Stanford en 1969

Chapitre 1 : Robotique de groupe

qui a été une plate-forme de démonstration des recherches en intelligence artificielle, le robot Hilare du LAAS a été construit en **1977** et le Genghis, développé par Rodney Brooks au MIT au début des années **1990**.

Le domaine de la robotique peut être présenté selon cinq grandes thématiques, fortement liées :

- la conception mécanique, électronique et informatique (« mécatronique ») ;
- l'action et le mouvement notamment la commande et les contrôles sensori-moteurs;
- la perception (multi-sensorielle) et la représentation de l'environnement (voir figure 1.5);
- la prise de décision, le raisonnement et l'apprentissage,
- l'interaction et la coopération avec d'autres robots ou avec l'homme.

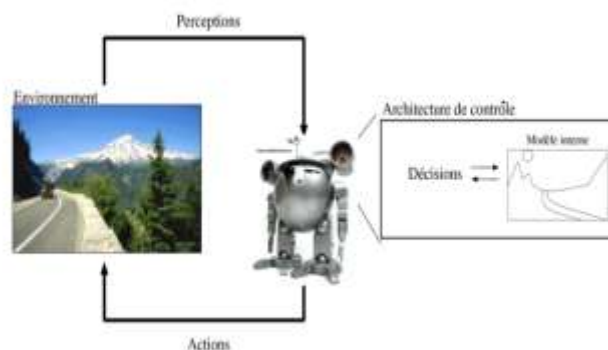


Figure 1.5 Interactions d'un robot avec son environnement : Selon les approches, un modèle interne de l'environnement peut être utilisé ou non.

Trois générations de robots peuvent être distinguées:

1. **Le robot est passif** : il est capable d'exécuter une tâche qui peut être complexe, mais de manière répétitive. Il ne doit pas y avoir de modifications intempestives de l'environnement et l'auto-adaptabilité est très faible.
2. **Le robot devient actif** : il devient capable d'avoir une image de son environnement et donc de choisir le bon comportement (sachant que les différentes configurations ont été prévues). Le robot peut se calibrer tout seul.
3. **Le robot devient « intelligent »** : le robot est capable d'établir des stratégies, ce qui fait appel à des capteurs sophistiqués et souvent à l'intelligence artificielle.

1.3.2. Architecture générale d'un robot

L'architecture générale d'un robot mobile est composée de trois principaux modules suivants (Figure 1.6) [19]:

- 1) **Module de perception** : ce module permet au robot de percevoir l'état de son environnement et des autres robots.
- 2) **Module de communication** : il permet l'échange d'informations utiles entre les robots.
- 3) **Module d'action** : le robot exécute l'action associée à un comportement donné correspondant à la situation dans laquelle il se trouve.

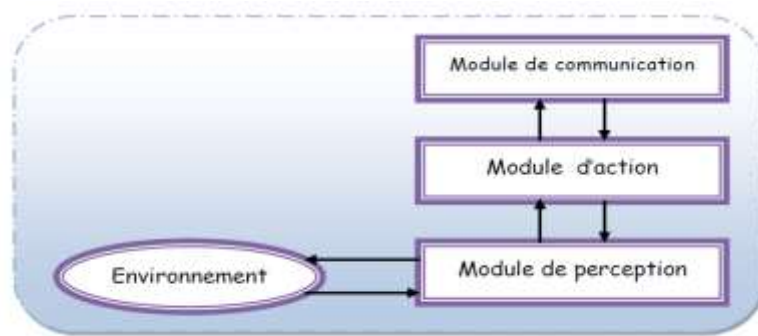


Figure 1.6 Architecture générale d'un robot mobile

1.3.3. Architecture de contrôle d'un robot mobile autonome

Drouilly [20] a distingué deux types principaux de contraintes pour l'autonomie des robots:

1. la capacité d'extraire des données de l'environnement (localiser des objets, des obstacles, le robot etc.) et
2. la capacité à traiter ces informations intelligemment (prendre des décisions, savoir comment interagir avec tel ou tel objet).

L'architecture de contrôle permet d'organiser les relations entre les trois grandes fonctions qui sont la perception, la décision et l'action. La classification des architectures de contrôle repose sur la stratégie utilisée pour contrôler le système et pour atteindre ses objectifs [15]. La figure 1.7 montre les principales architectures de contrôle utilisées dans la littérature. Il est à signaler que quelques auteurs se limitent au niveau un et ne distinguent pas l'architecture distribuée de l'architecture hiérarchique.

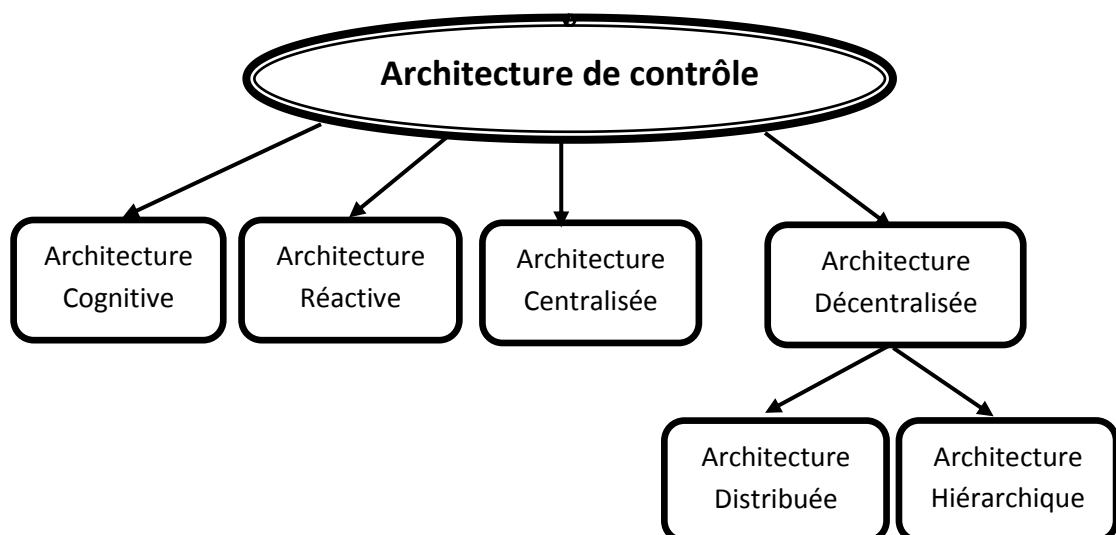


Figure 1.7 Classification d'architecture de contrôle

En se basant sur cette classification, on présente l'état de la recherche dans chaque type d'architecture.

1.3.3.1. Architecture de contrôle cognitive

Le principe de cette architecture repose sur la décomposition du contrôle en niveaux : perception, modélisation, planification, exécution des tâches et commandes de moteurs [15].

1.3.3.2. Architecture de contrôle réactive

Dans cette architecture, le fonctionnement est gouverné par la mise en place de réactions pour affronter les diverses situations auxquelles un robot mobile est confronté lorsqu'il évolue dans son environnement. Le lien « direct-mapping » entre les informations capteurs et les primitives d'actionnement doit être respecté pour un bon fonctionnement [21]. Braitenberg[22] a développé une machine « la machine de Braitenberg » ou le mappage direct entre les informations fournies par les capteurs et les actions des moteurs se fait via une relation du stimulus-réponse [15].

1.3.3.3. Architecture de contrôle centralisée

Dans une architecture centralisée, un agent de contrôle est responsable des informations globales concernant l'environnement ainsi que les robots. Cet agent de contrôle communique avec tous les robots pour permettre le partage des informations afin de réaliser les tâches [23]. De nombreux travaux ont investi dans ce type d'architecture; *Gerkey* [24] utilise une méthode de contrôle pseudo-centralisée qui permet une coopération efficace entre robots mobiles pour pousser un objet volumineux (système « pusher-wather »). Cette architecture possède un point de rupture dans le cas où le superviseur tombe en panne. Un autre inconvénient d'une telle architecture est la complexité du contrôle qui augmente en fonction du nombre de robots.

1.3.3.4. Architecture de contrôle décentralisée

Les architectures décentralisées se divisent en deux catégories : les architectures distribuées où il n'existe aucun agent de contrôle central, tous les robots sont autonomes et égaux en matière de contrôle et les architectures hiérarchiques où il existe un ou plusieurs agents de contrôle centraux, ces agents de contrôle sont locaux à des sous-parties de l'espace de travail appelées clusters. Les architectures décentralisées répondent mieux à la problématique des environnements inconnus ou variables et ont de meilleurs résultats en termes de fiabilité, de flexibilité, d'adaptabilité et de robustesse [25]. Parmi les architectures développées ; l'architecture *KAMARA* (Karlsruhe Multi Agent Robot Architecture) proposée par Laengle qui utilise un mécanisme de contrôle distribué [23]. Concernant les systèmes multi-robots utilisant ce type d'architecture, *M+* et *ALLIANCE* qui utilisent une architecture distribuée [25] et le système *CEBOT* qui utilise une architecture hiérarchique dans laquelle les robots sont physiquement couplés avec d'autres robots pour former des clusters.

1.3.4. Domaines d'application

Chapitre 1 : Robotique de groupe

L'introduction des systèmes robotisés répond généralement à trois objectifs principaux : réaliser les travaux dangereux (tout en limitant l'exposition de l'homme), effectuer des tâches que les robots individuels ne peuvent pas effectuer et accomplir des tâches pénibles ou répétitives. Ces objectifs opérationnels couvrent un large spectre de missions telles que l'observation, la reconnaissance, la surveillance de zones ou de frontières, la protection des itinéraires ou la logistique [26].

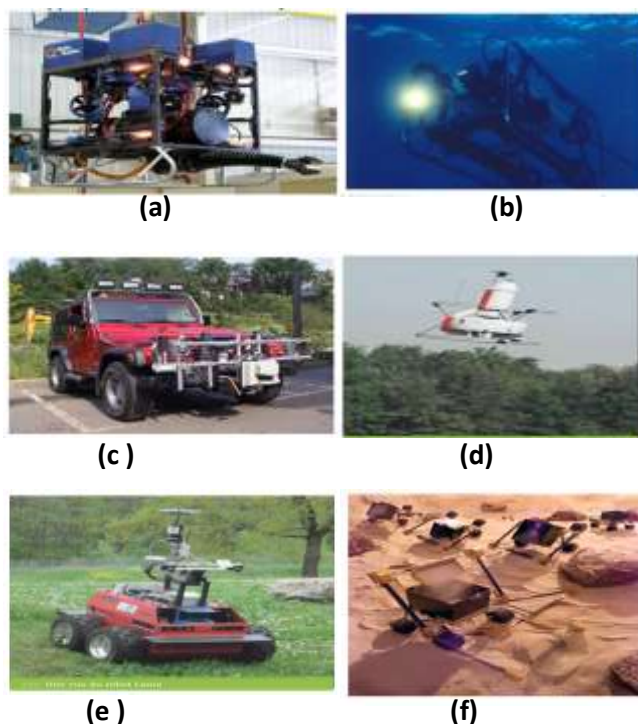


Figure 1.8 Domaines d'application de robotique

(a): le ROV3 développé par des chercheurs d'Hydro-Québec,
(b): Les véhicules sous-marins commercialisés par DeepOcean Engineering,
(c): véhicules routiers autonomes,
(d):drones intelligents,
(e) et (f) : robots d'exploration spatiale

1.3.5. Réseaux de robots

1.3.5.1. Définition

Un réseau de robots est un réseau radio composé de nœuds mobiles qu'on peut commander. Les nœuds du réseau peuvent être homogènes ou hétérogènes et peuvent inclure des robots mobiles, des capteurs statiques et/ou des capteurs mobiles. Le réseau de robot a atteint un niveau de complexité considérable dans tous les domaines terrestre, marin, aérien voire spatial. Cette complexité provient notamment de l'intégration de fonctionnalités multiples : perception avancée (fondée par exemple sur la fusion multi-capteurs), planification, navigation, commande, comportements autonomes, etc., sans omettre celles relatives à la téléopération et à la coopération de robots. Le comité technique IEEE a adopté la définition suivante d'un réseau de robots [14]:

« A networked robot is a robotic device connected to a communications network such as the Internet or local area network (LAN). The network could be wired or wireless, and based on any of a variety of protocols such as the transmission control protocol (TCP), the user datagram protocol (UDP), or 802.11 »

Chapitre 1 : Robotique de groupe

« Un réseau de robot est un dispositif robotique connecté à un réseau de communication tel que l'internet ou un réseau local (LAN). Le réseau peut être filaire ou sans fil repose sur une variété de protocoles tels que le protocole de transmission de contrôle (TCP), le protocole de datagramme utilisateur (UDP) ou 802.11 »

Kumar et al [14] ont distingué deux types de réseaux de robots ; *le réseau téléopéré* « *Teleoperated network* » (voir figure 1.9) où les superviseurs humains envoient des commandes aux robots et reçoivent leurs rétroactions via le réseau. Et *le réseau autonome* « *Autonomous network* » (voir figure 1.10) où les robots travaillent sans aucun contrôle.

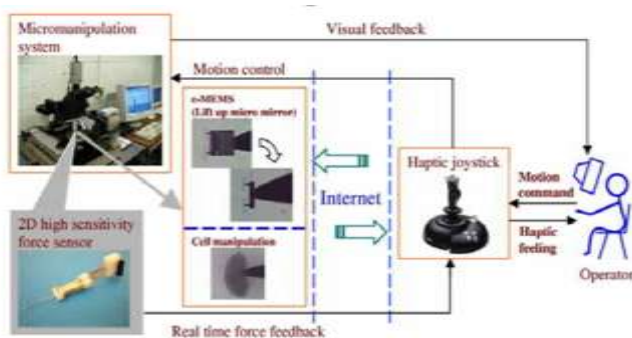


Figure 1.9. Réseau téléopéré



Figure 1.10. Réseau de robots autonome

1.3.5.2. Communication

La communication dans un réseau de robots est un point crucial pour l'accomplissement des tâches. Les robots communiquent les uns avec les autres, partageant la même mission, naturellement grâce à la communication sans fil afin de coordonner, d'auto-organiser et de coopérer. La communication sans fil est un excellent candidat pour l'échange d'informations inter-robot. Ainsi, les robots en essaim combinés avec la communication sans fil a été une force essentielle au cours des dernières années et a actuellement étendu aux réseaux multi-sauts sans fil, qui comprennent les réseaux radio ad hoc, réseaux de capteurs et les réseaux maillés sans fil. Pour que les robots coopèrent d'une façon plus rapide et plus efficace, les technologies de communication des réseaux de robots ont été développées. La technologie infrarouge a été la première technologie appliquée dans la communication sans fil de robots en raison de son coût faible. Cependant, cette technologie n'est pas toujours efficace et fiable pour la communication surtout à la présence des obstacles. La technologie de radio fréquence a été utilisée pour permettre aux robots de communiquer via le lien de point-à-point de fréquence radio ou via un mécanisme de diffusion. FHSS (The frequency hop spread spectrum) et DSSS (direct sequence spread spectrum) sont des technologies de modulation largement appliquées à l'industrie scientifique médicale (ISM). La prolifération de l'Internet a motivé la recherche à utiliser le réseau local sans fil « LAN » (IEEE 802.11), les normes Bluetooth et les réseaux ad hoc dans les systèmes de robots mobiles [27]. Ces trois technologies sont montrées dans la figure 1.11 suivante :

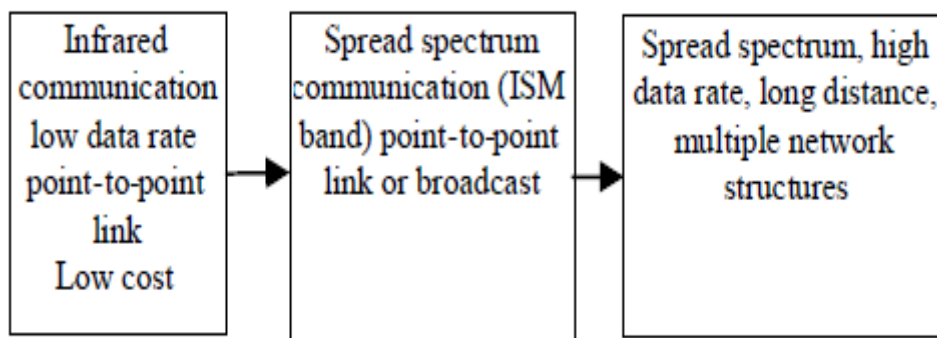


Figure 1.11 Évolution des technologies de communication de robots sans fil

Le tableau suivant (**Tab1.1**) présenté par [27] compare entre les différentes technologies de communication des robots mobiles. La comparaison est faite selon le critère de la bande de fréquence radio, le type de modulation, le taux de transmission, la portée de communication, la structure réseau et la puissance de signal de chaque technologie.

	Infrared	IEEE 802.11b	Bluetooth
RF band (GHz)		2.4/5	2.4/5
Modulation		DSSS	FHSS
Data rate (Mbps)	0.1- 4	11 (shared)	0.72 future 10/20
Range (m)	4	30-1000	10-100
Network structure	PPP	Infrastructure and ad hoc	Ad hoc
Power	5 mw	≤ 1 w	0.3mw-30mw

Tab1.1 Tableau comparatif entre les technologies de communication des robots

Les robots ont besoin de communiquer pour coopérer effectivement. Par cette interaction, les robots peuvent d'une part partager les informations de position, l'état de l'environnement et les données des capteurs avec les autres dans le réseau. Selon le mode d'interaction, la communication peut être explicite ou implicite [25] (voir figure 1.12).

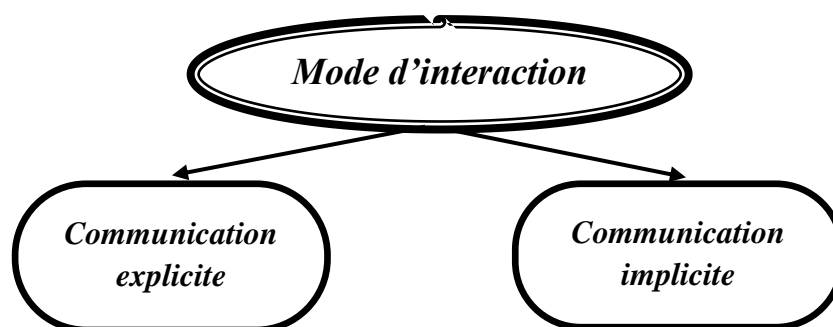


Figure 1.12 Classification de la communication selon le mode d'interaction

- **La communication explicite** consiste en une opération relationnelle entre une entité et une ou d'autres entités [28]. L'envoi, la réception et l'interprétation des messages nécessitent le recours à des langages de communication [15]. Le langage *KQML* "Knowledge Query Manipulation Language" développé en 1993 par le consortium DARPA-KSE (Knowledge Sharing Effort) et le langage *FIPA-ACL* "Agent Communication Language" proposé en 1997 par la FIPA (Foundation for Intelligent Physical Agents) sont les plus répandus [15].
- **La communication implicite** (« indirecte » ou « through the world ») se réfère au mode selon lequel un robot reçoit des informations d'autres robots à travers l'environnement [25] et cela à l'exemple des fourmis qui déposent des phéromones dans l'environnement pour guider leurs congénères vers les sources de nourriture. Cette communication peut être aussi illustrée par ce qui est appelé par Paul Grassé en 1959, la « *stigmergie* » [21]. Elle a prouvé son efficacité pour sélectionner des routes dans les réseaux naturels et artificiels de transport ou dans les réseaux de télécommunications.

Une autre classification est présentée dans la figure 1.13. Le type de communication peut être une communication entre robots ou une communication entre un utilisateur et le réseau de robots [28].

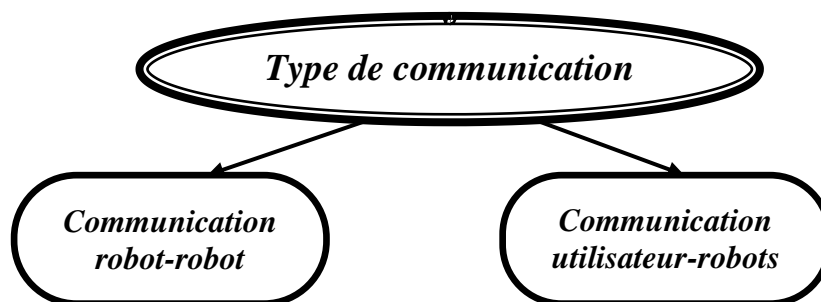


Figure 1.13 Classification des types de communication

Concernant la communication entre robots, un travail intitulé "*Hormone-Inspired Adaptive Communication and Distributed Control for CONRO Self-reconfigurable Robots*" [29] peut être classé dans les recherches de ce système ainsi que dans les systèmes d'inspiration biologiques et les systèmes de contrôle. Les modules *CONRO* sont des micro-robots qui peuvent se connecter physiquement et de manière autonome entre eux de façon à former différentes configurations : chaînes, arbres, cercles, polygones, etc. Un protocole appelé "Adaptive Communication" autorise les micro-robots *CONRO* à découvrir continuellement les changements dans leur topologie. Ce travail prend comme modèle la génération d'hormones utilisées comme moyen de communication et de contrôle, une hormone qui peut se propager dans l'ensemble du réseau de modules et provoque des réactions différentes selon les modules en fonction de leur état interne et des informations locales. *Shenet al.*[29] montrent que l'auto-

reconfiguration des robots demande une nouvelle méthodologie de communication et de coopération et que le concept biologique d'hormone peut être une source d'inspiration de plusieurs idées.

Pour la communication utilisateur-robots, plusieurs auteurs ont investi ce type de communication et plusieurs systèmes ont été développés. Le système *MAGIC2* [30], élaboré pour le contrôle opérationnel des véhicules aériens sans pilote combine les panneaux de contrôle, mais semble être limité à un maximum de quatre véhicules par l'opérateur. Jones et al[30] ont décrit des dialogues entre l'humain et le système. L'application visée est l'utilisation d'une équipe de robots dans la construction spatiale. Les dialogues sont conduits par l'utilisateur avec une communauté d'agents par des séries de questions implicites et explicites sur l'état des robots et de l'environnement. L'opérateur joue un rôle important dans la gestion des ressources et la planification des tâches des robots. La notion d'*affordance* développée par les auteurs fournit les types d'opérations et de manipulations qui peuvent être effectués sur les objets ou éléments matériels ou logiciels qui entrent dans la composition du système [28]. Fong[31] a développé un système d'interaction nommé « Human-Robot Interaction Operating System : (HRI/OS) ». Le HRI/OS fournit un logiciel pour construire une équipe humain-robot, qui permet aux humains et aux robots d'engager dans des tâches axées dialogue et de faciliter l'intégration de robots via une API extensible.

1.3.5.3. Coopération, coordination et collaboration

Un réseau de robots est caractérisé par une meilleure efficacité, robustesse et faisabilité par rapport à un robot solitaire pendant la réalisation des tâches complexes. Les tâches telles que l'exploration ou la cartographie peuvent, en principe, être effectuées plus rapidement avec une augmentation du nombre de robots. Plus de rapidité dans l'accomplissement des tâches peut être obtenue par le déploiement de plusieurs robots effectuant des opérations en parallèle, mais d'une manière coordonnée. Donc, les robots de réseau mettent en œuvre des interactions complexes, comme la *coopération*, la *coordination*, la *collaboration*, la *compétition* et la *négociation*. Bourjoit [32] et David [33] ont distingué entre ces concepts :

- On parle de **coopération** lorsque les robots travaillent ensemble pour atteindre un but commun. Noter qu'une coopération efficace entre robots nécessite un bon degré de coordination (**exemple** : les membres d'une même équipe de football « soccer robots »)
- La **coordination** consiste à organiser les activités liées à la résolution d'un problème de manière à éviter les interactions néfastes et à exploiter les interactions bénéfiques. **Exemple** : le cas d'une équipe de robots devant explorer la planète Mars ou encore des véhicules autonomes se déplaçant dans de mauvaises conditions climatiques.
- Les robots sont dans une situation de **collaboration** s'ils partagent un même but de façon intermittente.

Chapitre 1 : Robotique de groupe

- On parle de **compétition** quand les robots du réseau ont des buts incompatibles (**exemple** : les deux équipes dans un match de football)
- La **négociation** a pour objectif la gestion des conflits entre agents, c'est-à-dire qu'elle consiste à aboutir à un accord acceptable par l'ensemble des agents impliqués.

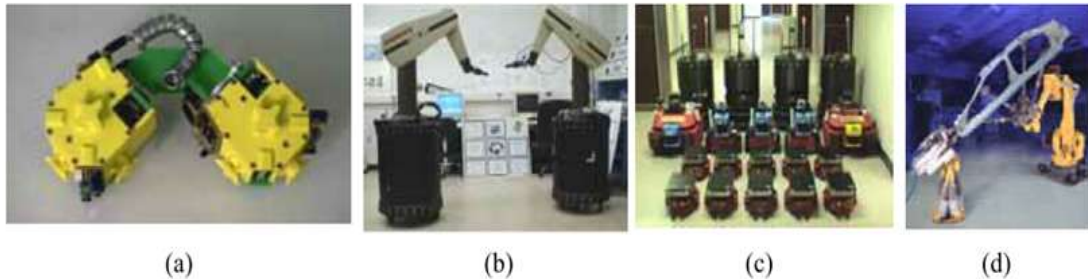


Figure 1.14 Petits modules: ils peuvent se connecter automatiquement et communiquer des informations pour effectuer des tâches de déplacement (a); bras de robot sur des bases mobiles peuvent coopérer pour accomplir des tâches ménagères (b); essaims de robots peuvent être utilisés pour explorer un environnement inconnu (c), et des robots industriels peuvent coopérer dans les opérations de soudage (d)

1.3.5.4. Domaines d'applications

Plusieurs applications ont été développées récemment dont le but est d'étudier plusieurs problèmes fondamentaux dans les réseaux de robots (par exemple le rassemblement, la dispersion et le déplacement en formation). Il existe de nombreuses démonstrations réussies de réseaux de robots. L'importance et l'impact potentiel des réseaux de robots sont apparus dans les exemples suivants :

Un exemple d'un projet avec des véhicules hétérogènes dans un cadre intérieur est représenté dans la figure 1.14 dans laquelle les robots se déploient pour former un réseau de capteurs pour détecter les intrus. Ce projet a démontré la capacité de déployer 70 robots afin de détecter les intrus dans un bâtiment inconnu (Université du Tennessee, l'Université de Californie du Sud et SAIC) [34].



Figure 1.15 Déploiement des robots dans un environnement inconnu pour détecter les intrus

Schwager et al[35] présentent une stratégie de contrôle qui permet à un groupe de robots mobiles de se positionner pour optimiser la mesure de l'information sensorielle dans l'environnement tout en agrégeant dans les zones de fort intérêt. En outre, les robots ne savent pas à l'avance où sont les zones d'intérêt sensoriel, mais ils apprennent cette information à partir des mesures de capteurs en ligne. Selon

Chapitre 1 : Robotique de groupe

Schwager et al, le contrôleur peut être utilisé par des groupes de robots pour effectuer des tâches telles que la surveillance environnementale et le nettoyage, la surveillance automatique des pièces, des bâtiments et des villes, ou la recherche et le sauvetage. Les auteurs ont donné un exemple où ils ont considéré une équipe de robots chargée de nettoyer un déversement de pétrole. Le contrôleur dans ce cas permet aux robots de disperser sur le déversement, apprendre les zones où le déversement est le plus profus et de concentrer leurs efforts sur ces domaines, sans négliger les zones où le déversement est rare.

Mandal et al [36] ont implémenté une technique de couverture de terrain en utilisant un réseau de robots en essaim ; ce type de couverture peut être utilisé dans les fermes et les zones ouvertes. Les auteurs dans cet article présentent comment les robots peuvent être utilisés pour augmenter le parallélisme et réduire le temps de couverture du terrain.

Mellouk et al [37] ont présenté « le réseau de robots ubiquitaire » qui est un nouveau domaine de recherche dans les systèmes autonomes intelligents. Il vise les nouvelles applications innovantes dans lesquelles les systèmes robotiques sont intégrés dans les environnements informatiques ubiquitaires comme des entités autonomes physiques. Ces entités sont capables d'interagir de manière autonome avec l'environnement ambiant et de fournir des services au plus. À l'avenir, les robots comme des entités cognitives seront capables de coordonner leurs activités avec d'autres entités physiques ou logiques, de se déplacer, de capturer, d'explorer le milieu ambiant et de décider et d'agir pour répondre aux situations auxquelles ils peuvent être confrontés.

Dans l'industrie de soins de santé, les réseaux permettent aux professionnels de la santé d'interagir avec leurs patients, d'autres professionnels, des instruments de diagnostic coûteux et dans l'avenir avec des robots chirurgicaux. La télémédecine représente un essor de croissance important dans les dispositifs travaillant à distance dans un réseau. Ils remplaceront les dispositifs médicaux d'aujourd'hui [14].

La surveillance de l'environnement est une application clé pour les réseaux des robots. En exploitant la mobilité et la communication, l'infrastructure robotique permet l'observation et la collecte de données à des échelles plus élevées dans les divers aspects de la surveillance écologique. Ceci est important pour les politiques de réglementation de l'environnement (par exemple, nettoyage de l'air et de l'eau) [14].

De nombreux projets de recherche se penchent sur les comportements de groupe ou d'intelligence collective en réalisant des comportements collectifs observés dans la nature. Par exemple, l'Union européenne (UE) a plusieurs projets coordonnés à l'échelle européenne sur l'intelligence collective ou l'intelligence en essaim. Le projet I-Swarm à Karlsruhe et le projet Swarm-Bot à l'École Polytechnique Fédérale de Lausanne (EPFL) sont des exemples de projets s'intéressant à l'intelligence en essaim. Le Laboratoire d'Analyse et d'Architecture des Systèmes (LAAS) a un groupe solide dans la robotique et l'intelligence artificielle. Ce groupe a eu une longue histoire de recherche fondamentale et appliquée dans les réseaux de robots. L'intégration de

Chapitre 1 : Robotique de groupe

plusieurs véhicules sans pilote pour des applications telles que la cartographie du terrain et la lutte contre l'incendie.

Pour pouvoir juger que le réseau de robots a réalisé une tâche donnée avec succès et d'une façon parfaite, plusieurs points doivent être mis au point. Par exemple *Reich et al* [38] ont montré qu'un système de recherche et de sauvetage entièrement sélectionné doit localiser rapidement et précisément les victimes, construire une carte de l'espace de recherche, maintenir la communication avec l'opérateur distant et évaluer l'état de la victime et de contribuer aux efforts de secours.

1.4. Conclusion

Les biologistes ont été les premiers à étudier les animaux et à essayer de comprendre leur comportement. Un bon nombre d'espèces jugées moins "intelligentes" sont capables de résoudre des problèmes complexes sans rapport hiérarchique sans planification prévue à l'avance, mais pourtant d'une efficacité et d'une robustesse implacable. Ces comportements d'Action-Reflexe sont à la base des comportements réactifs utilisés dans le monde des réseaux de robots.

Dans ce chapitre, Tout d'abord, nous avons fait un rappel sur le domaine de l'intelligence en essaim en expliquant son principe de fonctionnement. Ensuite, nous avons présenté un bref historique de la robotique. Nous avons abordé aussi l'architecture générale ainsi que l'architecture de contrôle d'un robot mobile autonome. La dernière partie à était consacrée aux réseaux de robots où nous les avons définis. Nous y avons aussi expliqué les notions de communication, coopération, collaboration et coordination tout en montrant leur rôle fondamental dans la mise en œuvre d'un réseau de robots.

2

CHAPITRE

Réseaux sans fil

2.1. Introduction

Ces dernières années, plusieurs standards et technologies de communication sans fil ont émergé et se sont imposés dans notre vie quotidienne. Un réseau sans fil est un réseau dans lequel au moins deux terminaux (téléphones, ordinateurs portables, tablettes numériques, etc.) peuvent communiquer sans liaison filaire. Grâce aux réseaux sans fil, un utilisateur peut rester connecté tout en se déplaçant dans un périmètre géographique plus ou moins étendu. Les réseaux sans fil sont basés sur une liaison utilisant des ondes électromagnétiques au lieu de placer des câbles habituels [39]. La majorité de ces réseaux sans fil reposent sur des infrastructures centralisées et des paradigmes de communication de type client/serveur ou maître/esclave. En effet, ces réseaux utilisent la notion de points d'accès ou stations de base afin d'assurer la connectivité entre des réseaux distants. Afin de s'affranchir de ces architectures centralisées qui sont très coûteuses (en termes de temps d'accès et d'énergie), un nouveau type d'architecture de réseau est apparu, connu sous le nom de réseau ad hoc supporté par un groupe de travail de l'IETF (Internet Engineering Task Force)[40]. Les réseaux Ad hoc sont considérés comme une généralisation ultime des réseaux sans fil, car ils limitent au maximum le rôle de l'infrastructure fixe. Puisque la propagation radio des entités d'un réseau ad hoc est limitée, un protocole de routage est nécessaire afin de faire communiquer deux entités quelconques dans le réseau. Les protocoles de routage traditionnels MANET comme AODV, DSR et DSDV font l'hypothèse que le graphe du réseau est entièrement connecté et ne parviennent pas à envoyer les messages s'il n'y a pas un chemin complet de la source vers la destination au moment de l'envoi. Par conséquent, une nouvelle stratégie de routage a été proposée pour livrer le paquet de données avec succès dans un tel réseau fréquemment déconnecté dans l'espoir que l'un des nœuds intermédiaires sera à un moment donné à la portée du destinataire. Cette stratégie de routage est appliquée dans des réseaux dis tolérant au délai (Delay Tolerant Network). Ce type de réseau fonctionne dans les environnements de communication soumis à des retards et des perturbations.

Dans ce chapitre, nous allons décrire le principe des réseaux Ad hoc ainsi que les protocoles existants pour ce type de réseaux, ensuite nous allons introduire les réseaux tolérants au délai en expliquant leur principe de fonctionnement, leurs différents protocoles de routage et les domaines d'applications pour les deux types de réseaux.

2.2. Les réseaux sans fil

Un réseau sans fil, comme son nom l'indique, est un réseau composé de terminaux pouvant être fixes ou mobiles et communiquant sans liaison filaire. Contrairement à un réseau classique, les nœuds d'un réseau sans fil utilisent les ondes radiofréquences ou infrarouges pour communiquer. Les réseaux sans fil offrent une grande flexibilité d'emploi; en particulier, ils permettent la mise en réseau de sites dont le câblage serait

Chapitre 2 : Réseaux sans fil

trop onéreux à réaliser [41], voire impossible dans certaines régions où l'accès est difficile.

2.2.1. Classification des réseaux sans fil

Les réseaux sans fil peuvent être classés selon la zone de couverture (ou selon l'existence d'une infrastructure de communication (voir **Figure 2.1**).

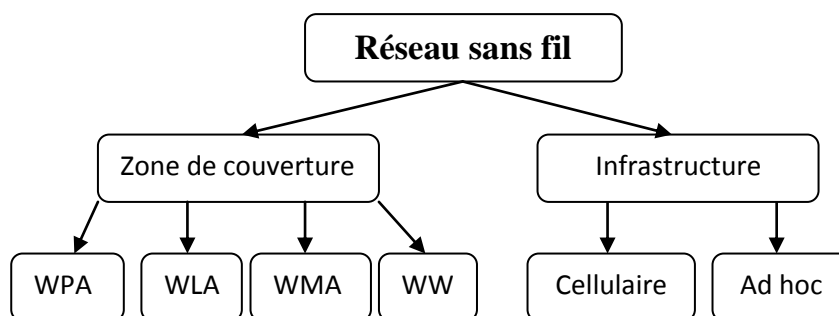


Figure 2.1 Classification des réseaux sans fil

A. Suivant la portée (zone de couverture)

En fonction de cette caractéristique, les réseaux sans fil peuvent être classés en quatre catégories ; celles-ci sont définies en fonction du périmètre géographique offrant une connectivité, appelée zone de couverture (voir **Tab 2.1**) [42]:

1. **Les réseaux personnels sans fil** (WPAN: Wireless Personal Area Network) technologie Bluetooth, infrarouge.
2. **Les réseaux locaux sans fil** (WLAN: Wireless Local Area Network) : technologie Wifi. (cette technologie est utilisée dans les réseaux Ad hoc).
3. **Les réseaux métropolitains sans fil** (WMAN : Wireless Metropolitan Area Networks) technologie BLR (Boucle Local Radio).
4. **Les réseaux étendus sans fil** (WWAN: Wireless Wide Area Networks) technologie GPRS.

Type	Couverture géographique	Performances	Normes
WPAN	Quelques dizaines de mètres	Modérées	Bluetooth (2 Mbps), IrDa(4 Mbps)
WLAN	Quelques centaines de mètres	Hautes	Wi.Fi (11.54 Mbps), HiperLAN (54 Mbps)
WMAN	Dans les limites d'une ville	Hautes	WiMAX (1 Mbps .10 Mbps)
WWAN	Mondiale	Faibles	GSM (9,6Kbps), GPRS (115 Kbps), UMTS (2Mbps)

Tab2.1 Comparaison entre les différents types de réseaux sans fil

B. Suivant l'infrastructure

Les réseaux sans fil peuvent être classés en deux catégories selon l'existence ou non de l'infrastructure. Dans notre travail nous nous sommes intéressés aux réseaux sans fil mobiles. Donc, nous pouvons distinguer, les réseaux mobiles avec infrastructure de communication et les réseaux mobiles sans infrastructure de communication ou les réseaux Manet.

B1. Réseau mobile avec infrastructure

En mode avec infrastructure, également appelé le mode BSS (Basic Service Set) certains sites fixes, appelés station support mobile (MobileSupport Station) ou station de base (SB) sont munis d'une interface de communication sans fil pour la communication directe avec des sites ou unités mobiles (TM), localisés dans une zone géographique limitée, appelée cellule (**Figure 2.2**).

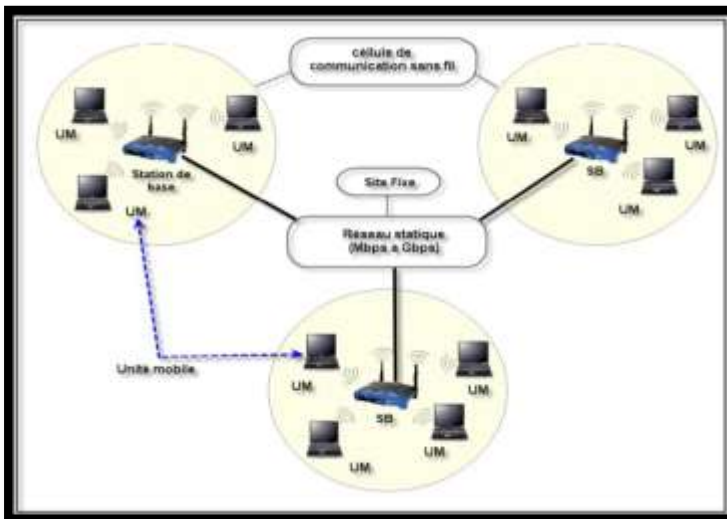


Figure 2.2. Modèle d'un réseau mobile avec infrastructure : chaque ordinateur (appelé aussi unité mobile) se connecte à un point d'accès (appelé aussi site fixe ou station de base SB) via une liaison sans fil. L'ensemble formé par le point d'accès et les unités mobiles situées dans sa zone de couverture, est appelé ensemble de services de base (en anglais *Basic Service Set*, noté BSS) et constitue une cellule.

B2. Réseau mobile sans infrastructure (MANet)

Le réseau sans infrastructure, appelé réseau Ad hoc, ne comporte pas une station de base assurant la communication entre les entités (terminaux), tous terminaux du réseau communiquent d'une manière directe en utilisant leurs interfaces de communication sans fil (**Figure 2.3**). L'absence de l'infrastructure ou du réseau filaire composé des stations de base, oblige les unités mobiles à se comporter comme des routeurs qui participent à la découverte et la maintenance des chemins pour les autres hôtes du réseau[43].

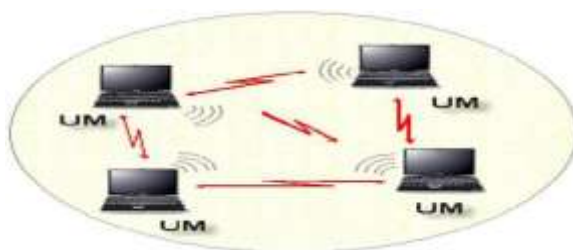


Figure 2.3 Modèle d'un réseau mobile sans infrastructure : Des unités mobiles communiquent entre eux sans infrastructure en utilisant des liens sans fil.

2.2.2. Réseau Ad hoc

2.2.2.1.Introduction

Un réseau ad hoc peut être défini comme une collection d'entités interconnectées par une technologie sans fil formant un réseau temporaire sans l'aide de toute administration [44] (voir **Figure 2.4**). Les réseaux ad hoc se forment de manière non planifiée et imprévisible, mais dans les configurations généralement étudiées leur composition et leur topologie évoluent la plupart du temps très peu. On peut alors les utiliser de la même façon que des réseaux statiques moyennant la mise en place d'un protocole de routage dédié tel qu'OLSR ou AODV. Une fois constitué, un réseau ad hoc a une composition relativement stable qui peut être conservée durant toute la durée de l'événement pour lequel il a été créé. Par exemple, les applications des réseaux ad-hoc, la mise en place très rapide de réseaux de communication dans les aéroports, hôtels ...etc. où les espaces peu équipés en infrastructures de communication lourdes. Par contre, les MANETs (Mobile Ad hoc NETWORKs) sont le type de réseau le plus instable en raison de la mobilité de leurs nœuds. Ils ne comportent pas d'infrastructure et les terminaux peuvent apparaître ou disparaître à tout moment suivant leurs capacités à communiquer ou leurs intérêts applicatifs. Par exemple, dans le cas des VANets (Vehicular Ad hoc Networks), sous-ensemble des MANets dans lesquels les nœuds sont des véhicules ou même dans le cas des RANets (Robot Ad hoc Networks) où le réseau est composé d'un ensemble de robots [45]. Dans les réseaux MANET les modèles de mobilité utilisés sont étudiés largement [46][47] ainsi que les effets de la mobilité sur les performances des réseaux. On trouve dans ce contexte le travail de HAQIQ et al[5] qui ont étudié l'effet de la mobilité et des modèles de trafic sur la consommation d'énergie dans les protocoles de routage MANET.

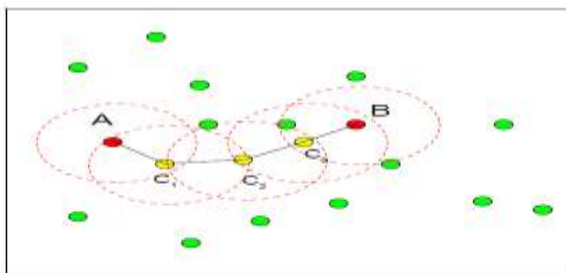


Figure 2.4 Réseau Ad hoc : Les points colorés réparties aléatoirement sont des nœuds. Les grands cercles en pointillés représentent les portées de leur interface radio. Pour aller de A à B (qui est hors de portée de A) les paquets passent par les nœuds intermédiaires C1, C2 et C3.

2.2.2.2.Piles protocolaires

A. Pile protocolaire des réseaux mobiles Ad hoc

Comme la plupart des réseaux filaires ou mobiles, les réseaux mobiles Ad hoc sont organisés selon une structure en couches. Cette dernière se décompose pour les réseaux mobiles Ad hoc en cinq couches (couche physique, couche liaison de données, couche réseau, couche transport et couche application) (voir **Tab2.2**).

<i>Réseaux filaires</i>		<i>Réseaux sans fils</i>
OSI	TCP /IP	Ad hoc
Couche application	Couche application (HTTP, FTP)	Couche application
Couche présentation		
Couche session	Couche Transport	Couche transport
Couche transport		
Couche réseau	Couche Internet (IP)	Couche réseau
Couche liaison	Couche interface Réseau	Couche de liaison de données (MAC et LLC)
Couche physique		Couche physique

Tab2.2 Représentation de la structure en couches des réseaux mobiles Ad hoc par rapport à OSI et TCP/IP

Dans les réseaux mobiles Ad hoc, la couche physique définit la modulation des ondes radioélectriques et les caractéristiques de la signalisation pour la transmission de données, tandis que la couche liaison de données définit l'interface entre le bus de la machine et la couche physique, notamment une méthode d'accès près de celle utilisée dans le standard Ethernet et les règles de communication entre les différentes stations. En ce qui concerne les couches plus hautes, elles maintiennent des comportements similaires à ceux de leurs équivalents TCP/IP [49].

B. Pile protocolaire de réseau de robot

Dans les réseaux de robots la communication entre les robots est d'une importance cruciale. Chaque robot doit échanger les informations récoltées via ses capteurs et de partager ses tâches avec les autres robots. Les robots sont souvent équipés des interfaces sans fil de faible puissance et de courte portée ce qui permet seulement la communication directe avec leurs voisins proches. Des robots avec ces caractéristiques ne peuvent pas communiquer via des liens de communication avec une connectivité dynamique et des erreurs de transmission. Par conséquent, il est pratiquement impossible pour que chaque robot connaisse la topologie du réseau à tout instant. L'utilisation du réseau sans fil ad hoc est la seule approche pratique pour la distribution de commande, de contrôle et de détection.

Les réseaux de robots mobiles sans fil fournissent l'infrastructure réseau pour soutenir les besoins de la qualité de service (QoS) (bande passante, latence et fiabilité) des communications du robot. Ils doivent soutenir: la reconfiguration rapide (802.11, anneau à jeton), la gestion de mobilité (IP mobile, AODV), la gestion d'accord de niveau de service (SLA), et la qualité de service (protocole Internet mobile). La figure.2.5 donne un modèle en couches des réseaux de robots mobiles. Le modèle est composé des couches transport, réseau, liaison de données et une couche physique [50].

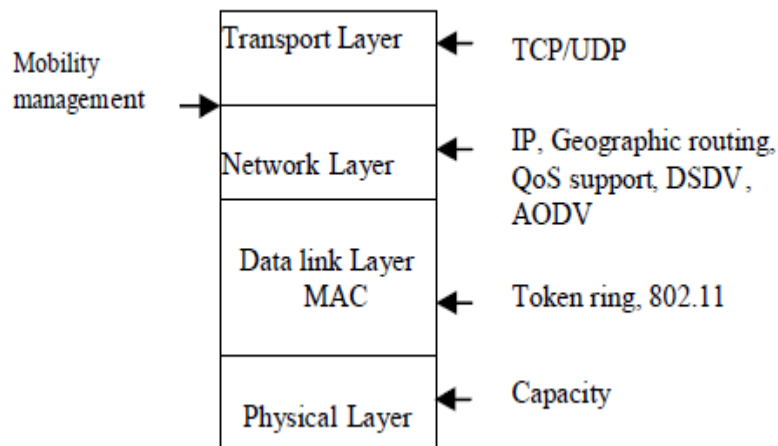


Figure 2.5 Représentation de la structure en couches des réseaux de robots mobiles

2.2.2.3. Routage dans les réseaux Ad hoc

A. Protocoles de routage dans les réseaux ad hoc

Aujourd'hui le réseau mobile ad-hoc est une technologie en plein essor, en raison de ses avantages tels que la topologie dynamique, la capacité de stockage et la bande passante limitée. Ces caractéristiques rendent le réseau attrayant pour certaines applications, mais en même temps elles posent des problèmes pour acheminer les paquets efficacement et précisément vers une destination particulière [51]. Le routage reste un défi dans les réseaux ad hoc et de nombreux protocoles de routage ont été proposés par les chercheurs. Ces protocoles ont été classés selon la manière de création et de maintenance des routes lors de l'acheminement des données en trois catégories distinctes qui sont les protocoles proactifs, les protocoles réactifs et les protocoles hybrides.

Les protocoles proactifs tels que l'OLSR (Optimized Link State Routing Protocol) ou DSDV (Destination- Sequenced Distance-Vector) maintiennent en permanence les routes vers les différents nœuds du réseau alors que les protocoles réactifs comme AODV (Ad hoc On Demand Distance Vector) ou DSR (Dynamic Source Routing) construisent une route « à la demande ». Afin de préciser la notion de protocole de routage pour les réseaux ad hoc, nous détaillerons, dans ce qui suit, trois protocoles parmi les plus répandus, AODV, OLSR et ZRP.

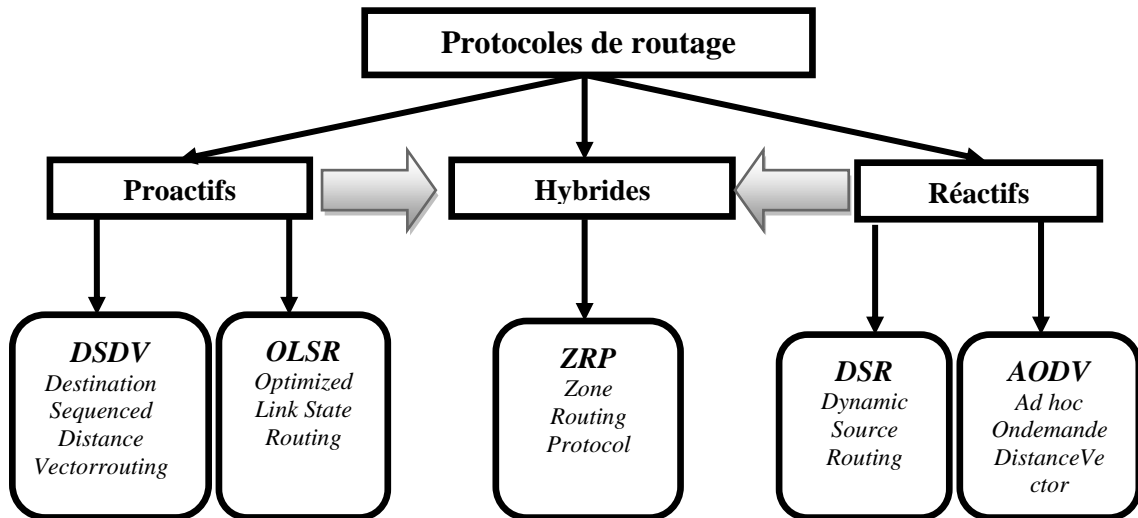


Figure 2.6 Classification des protocoles de routage dans un réseau Ad hoc

1. **Protocole de routage proactif** : un protocole de routage proactif est un protocole qui construit les tables de routage avant que la demande en soit effectuée. Il identifie en fait à chaque instant la topologie du réseau et échange périodiquement des paquets de contrôles [52].
2. **Protocole de routage réactif** : un protocole réactif est un protocole qui construit une table de routage lorsqu'un nœud en effectue la demande. Il ne connaît pas la topologie du réseau, il détermine le chemin à prendre pour accéder à un nœud du réseau lorsqu'on lui demande. La recherche sera globale [52].
3. **Protocole de routage hybride** : il existe aussi des protocoles hybrides qui sont à la fois proactif et réactif. L'objectif principal du protocole hybride est de minimiser la latence et la diffusion broadcast. Pour cela, le réseau est divisé en deux régions. Un nœud utilisera un protocole proactif pour le routage dans son voisinage proche et utilisera un protocole réactif pour le routage hors de la région précédente [52].

B. Les protocoles Ad hoc les plus adoptés

B1. Protocole de routage réactif: AODV

Le protocole de routage AODV (ou Ad hoc On demand Distance Vector), est un protocole de routage réactif dont les tables de routages sont mises à jour à la demande.

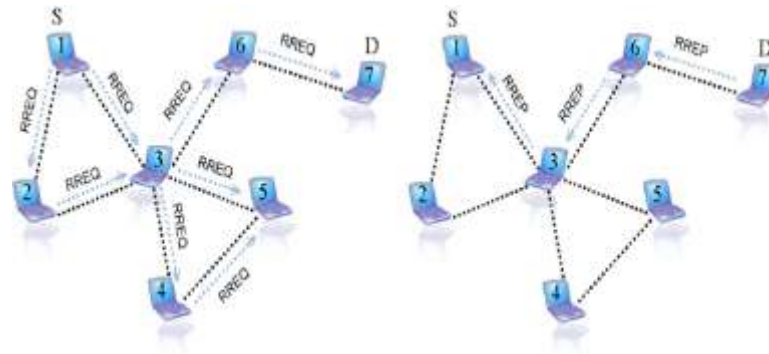


Figure 2.7 Le routage dans le protocole AODV

Dans AODV chaque nœud maintient une table de routage, dont chaque entrée contient des informations propres à une destination particulière. Chaque entrée contient aussi les éléments suivants : identificateur du destinataire, nombre de sauts nécessaire pour arriver au destinataire, le second saut menant vers le destinataire, un numéro de séquence destinataire (qui permet d'éviter les paquets inutiles et d'éviter les boucles). Un nœud diffuse une requête de route (RREQ : Route REQuest), dans le cas où il aurait besoin de connaître une route vers une certaine destination et qu'une telle route n'est pas disponible. Après la diffusion du RREQ, la source attend le paquet réponse de route (RREP : Route REPLY). Si ce dernier n'est pas reçu durant une certaine période (appelée RREP_WAIT_TIME), la source peut rediffuser une nouvelle requête RREQ. Quand un nœud intermédiaire reçoit un paquet pour le transmettre vers un destinataire, il vérifie sa table de routage pour connaître le second saut à prendre et puis il transmet le paquet vers ce nœud. Cette procédure est répétée jusqu'à l'arrivée du paquet à sa destination.

Leur avantage se trouve dans le fait qu'il n'y a pas de surconsommation de bande passante. En revanche, on peut s'inquiéter du délai nécessaire, avant l'envoi d'un message, pour trouver la route, aboutissant à une impossibilité de se connecter alors que le destinataire est bien là [52]. Grâce à ces avantages, plusieurs améliorations ont été proposées pour le protocole AODV, parmi les variantes proposés on retrouve, le MAODV (Multicast Ad hoc On-Demand Distance Vector) [53] qui est proposé pour la réalisation dynamique, l'auto-organisation et le routage multicast en demande entre les nœuds mobiles dans les réseaux mobiles ad hoc (MANET). Un autre protocole qui est le DZ-MAODV (*Dense Zone Multicast Adhoc On-demand Distance Vector*) est proposé par ZEBDI [54] afin d'économiser le nombre de messages échangés et de garantir une meilleure utilisation de la bande passante conjuguée avec une meilleure qualité de transmission. GUP et al [51] ont implémenté un protocole appelé Dynamic Manet on demand (DYMO) proposé par Perkins&Chakeres en 2009. DYMO fonctionne de manière similaire au protocole AODV et n'ajoute pas de modifications supplémentaires à la fonctionnalité existante. Mais, le fonctionnement est par ailleurs très simple, la seule différence est que le protocole DYMO ne supporte pas le message HELLO inutile et le fonctionnement est purement basé sur les numéros de séquence attribués à tous les paquets qui garantit le bon routage des paquets à la destination et maintenir des routes sans boucle similaires à celui de AODV et DSDV

Chapitre 2 : Réseaux sans fil

et il permet aux nœuds d'évaluer la fraîcheur des informations de routage. GUP et al ont cité les avantages suivants de ce protocole par rapport à d'autres protocoles: DYMO a un débit plus haut par rapport à AODV, DSR ; puisque plus de paquets de routage sont générés et livrés par DYMO que les autres protocoles. Le protocole est de rendement optimum quand le réseau est grand et montre une mobilité élevée. La table de routage de DYMO est relativement moins coûteuse en mémoire que l'AODV même avec la fonction d'accumulation de chemin.

B2. Protocole de routage proactif : OLSR

OLSR (*Optimized Link State Routing*) est un protocole proactif à état de liens. Afin de maintenir à jour les tables de routage, chaque nœud implémentant OLSR diffuse régulièrement des informations sur son propre voisinage. Ces informations sont suffisantes pour permettre à chaque nœud de reconstruire une image du réseau et de trouver une route vers n'importe quelle destination. Mais cette diffusion ne se fait pas par une simple inondation (où chaque nœud retransmet simplement chaque nouveau paquet qu'il reçoit) ; technique utilisant une connaissance "locale" de la topologie. Un nœud découvre ses voisins directs et ses voisins à deux sauts. Cette découverte basée sur les paquets "hello" où chaque nœud émet régulièrement un message contenant la liste des nœuds qu'il entend, message avec une durée de vie TTL = 1. Un nœud possède ainsi la liste des nœuds qu'il entend (les voisins à un saut), et aussi la liste des nœuds entendus par ceux-ci. OLSR optimise la diffusion grâce au système des relais multipoints (Multi-Points Relays : MPR). Chaque nœud choisit dans ses voisins directs un sous-ensemble minimal de nœuds qui lui permettent d'atteindre tous ses voisins à deux sauts (Figure 2.8) [49]. Cependant, le problème du choix des MPR est en fait un problème difficile car cela revient à trouver un ensemble dominant dans un graphe (ensemble de nœuds tel que tout sommet du graphe soit dans l'ensemble, soit voisin d'un nœud de l'ensemble).

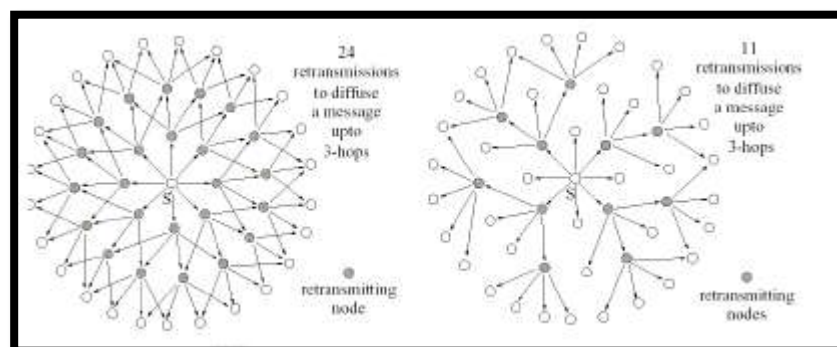


Figure 2.8 Le routage dans le protocole OLSR : cette figure montre l'optimisation obtenue après l'utilisation des MPR.

B3. Le protocole de routage hybride ZRP

ZRP (*Zone Routing Protocol*) définit pour chaque nœud un rayon (en nombre de sauts) au sein duquel les paquets sont routés en utilisant un protocole proactif [55]. ZRP est un protocole de routage dit hybride. Il met en place, simultanément, un routage proactif et un routage réactif, afin de combiner les avantages des deux approches. Pour ce faire, il passe par un concept de découpage du réseau en différentes zones, appelées "zones de routage". Une zone de routage pour un nœud, est définie par son "rayon de zone". Ce rayon correspond au nombre de sauts maximum qu'il peut y avoir entre deux nœuds (Figure 2.9).

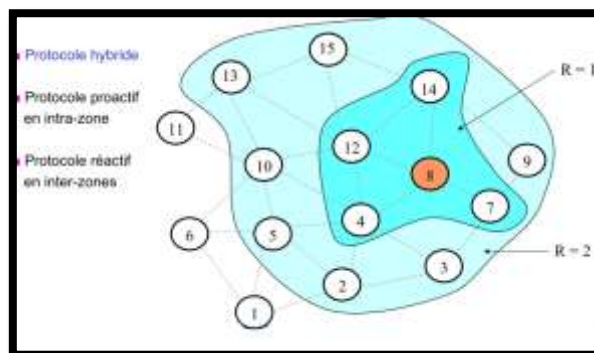


Figure 2.9 Le protocole hybride ZRP

C. Hiérarchisation du réseau

Les réseaux ad hoc sont structurés hiérarchiquement en différenciant les nœuds présents sur le réseau. Il est possible de voir la hiérarchisation des nœuds comme un arbre hiérarchique dans une entreprise. Les nœuds les plus hauts dans la hiérarchie dirigent les nœuds se trouvant en dessous. Les nœuds les plus bas dans la hiérarchie dépendent directement de leur supérieur hiérarchique. Ainsi, les nœuds se voient affectés des fonctions différentes suivant leur place dans la hiérarchie. Les réseaux ad hoc sont structurés hiérarchiquement sous forme de clusters.

Les nœuds étant en mouvement, un chef de groupe peut rompre la connectivité avec les nœuds qui lui sont rattachés. De même, il peut cesser de fonctionner, car la station peut s'éteindre ou ne plus avoir suffisamment d'énergie pour réaliser ce rôle. Lors de la perte de connectivité, cet ensemble de nœuds doit élire un nouveau chef de groupe. Pour cela, il suffit de réappliquer l'algorithme de plus faible identifiant ou de plus fort degré de connectivité sur l'ensemble des nœuds sans chef de groupe. Pour vérifier, la connectivité entre un chef de groupe et les nœuds qui lui sont rattachés, des paquets de contrôle sont échangés périodiquement. Nous pouvons citer dans ce contexte, les protocoles CGSR, HSR, CBRP, MMWN et LANMAR qui divisent le réseau en clusters et les protocoles OSR, PSR et NTDR qui divisent le réseau en BACKBONE.

Nous donnons le schéma (voir Figure 2.10) qui représente les réseaux de clusters, puis un autre schéma (voir Figure 2.11) qui montre un sous-ensemble de ce type de réseau qui est le réseau à BACKBONE [56].

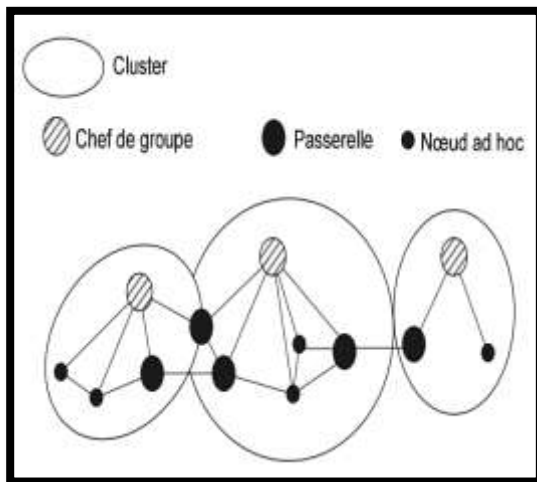


Figure 2.10 Topologie d'un réseau de clusters : Les réseaux de clusters sont un découpage du réseau en un ensemble de petits groupes gérés par des chefs de groupe.

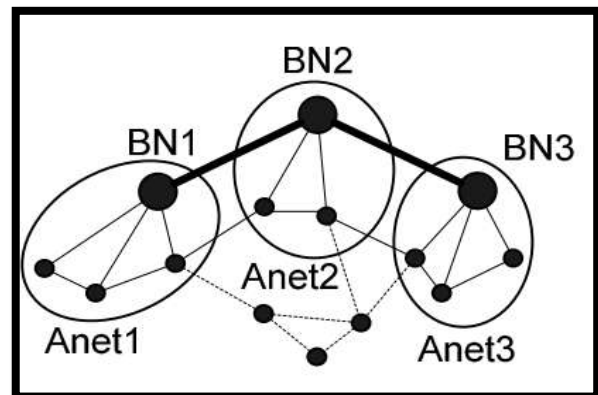


Figure 2.11 Topologie d'un réseau de BACKBONE : Un réseau mobile à BACKBONE a pour particularité d'avoir deux types différents de nœuds qui se différencient en fonction de leurs caractéristiques. Un réseau mobile de BACKBONE est composé d'un réseau de backbone (Bnet), d'un réseau d'accès (Anet) et d'un réseau ad hoc normal.

2.2.2.4. Problèmes du routage dans les réseaux MANET

La conception des protocoles de routage pour assurer la connexion des réseaux ad hoc est parmi les problèmes complexes dans ce type de réseaux, due à la mobilité des nœuds qui engendre le problème de déconnexion. D'autres défis peuvent être distingués dans l'acheminement des paquets, y compris l'évolutivité, l'efficacité du routage, l'adaptation à des réseaux sans fil de différentes densités. Il est nécessaire donc de prendre en compte les contraintes suivantes lors de toute conception de protocole de routage :

- 1. Minimisation de la charge du réseau :** deux sous problèmes doivent être considérés à savoir l'évitement des boucles de routage et l'empêchement de la concentration du trafic autour de certains nœuds ou liens.
- 2. Maintien de support fiable de communication multipoints :** la phase de maintenance des routes en cas des coupures doit être présente pour arriver à diminuer le temps de latence.
- 3. Optimalité de routage :** les chemins utilisés lors de routage des paquets doivent être optimaux en termes de différentes métriques de coûts (bande passante, nombre de liens, ressources du réseau, délais debout en bout, ...etc.).
- 4. Temps de latence :** plus que la connectivité augmente plus que les qualités de temps de latence et de chemins doivent être augmentés.

Un protocole de routage dans un Manet a pour but de trouver une route pour envoyer les données d'une source vers un destinataire donné à condition que cette route existe. L'efficacité des algorithmes de routage dépend de la qualité de connectivité du réseau. Pour cela, le problème de connectivité est largement étudié dans les Manet dû à la topologie dynamique qui caractérise ce type des réseaux. Plusieurs travaux focalisent sur le contrôle de la topologie comme étant la solution la plus efficace afin de maintenir la connectivité entre les nœuds. Un tel contrôle de topologie peut être réalisé en modifiant le rayon de transmission, la sélection d'un sous-ensemble donné des liens, ou le déplacement de certains nœuds comme il est présenté par Casteigts et al. [57]. D'autres solutions ont été proposées comme le travail de Basu [58] Zavlanos [59] pour le contrôle de mobilité et Tuan Le [60] dans la prise de conscience du réseau. Tous les travaux cités ci-dessus ont prouvé leurs efficacités dans le maintien de la connectivité au cours des différentes missions. Cependant, garder le réseau connecté restreint le mouvement des nœuds et par conséquent diminue les performances de la mission. Contrairement à ces travaux, nous examinons dans notre travail comment nous pouvons contrôler les déconnexions afin de parvenir à réaliser une tâche d'exploration par un réseau de robots.

2.2.2.5. Les applications des réseaux ad hoc

Les applications qui utilisent les réseaux ad hoc pour leurs communications sont caractérisées par une forte mobilité et ne reposent pas sur une infrastructure existante. Les réseaux militaires, les réseaux de capteurs mobiles et les réseaux d'urgence (incendies, tremblement de terre) sont des exemples de ces applications, car de pareil réseaux peuvent être utilisés pour assurer la liaison entre les différentes unités composant le réseau. Les réseaux ad hoc sont considérés aussi comme un des outils de base de l'informatique ubiquitaire (Ubiquitous Computing)[44].

2.2.3. Réseaux DTN

2.2.3.1. Introduction

Les réseaux traditionnels, comme Internet, sont basés sur l'hypothèse que certaines conditions doivent être vérifiées : un chemin debout en bout doit exister et être stable, les nœuds doivent toujours être alimentés en énergie, la bande passante doit être suffisante. Cependant, ces conditions sont très difficiles à obtenir dans certains scénarios, comme la liaison entre des satellites et le sol, la communication entre plusieurs planètes dans laquelle les délais de transmissions s'allongent ou dans les réseaux sans fil, lorsque les nœuds sont placés de façon très éparse et ne peuvent pas toujours communiquer entre eux. Pour parvenir à livrer les données malgré tous ces défis, Kevin Fall en 2002 a introduit le concept de DTN.

Les réseaux tolérants aux délais (Delay Tolerant Network) sont nécessaires pour connecter des hôtes lorsque la liaison est de mauvaise qualité ou présente des caractéristiques de latences et de bande passante telles que les protocoles usuels

(TCP/IP) échouent. Dans le contexte des réseaux interplanétaires (IPN), on doit pouvoir assumer des latences de plusieurs minutes où TCP accepte des latences de quelques secondes. Contrairement à Internet, ces réseaux peuvent supporter des délais plus importants et variables, de longues périodes de déconnexion, un taux d'erreur élevé et l'asymétrie de transfert de données [61]. Pour les propriétés qu'ils offrent, ces réseaux font également l'objet d'étude et d'utilisation par les réseaux dit « difficiles » (Challenge d'Inter networks) pour lesquels une connectivité permanente et debout en bout ne peut être garantie due à la couverture radio sans fil limitée, la dispersion étendue des nœuds mobiles et les contraintes d'énergie des ressources. Parmi les exemples de ces réseaux on cite ceux qui s'opèrent dans les réseaux mobiles ou dans des environnements extrêmes terrestres ou simplement comme prévu dans l'espace [62]. Le routage DTN est difficile en raison des durées longues et fréquentes de déconnexion. Ainsi, ce type de routage (DTN) permet de faire face aux problèmes de mobilité et d'énergie limitée des équipements de communication sans fil [61]. On constate aussi qu'il peut être exploité avec profit pour des applications de calculs distribués sur des sites mal connectés entre eux.

2.2.3.2. Définition des réseaux DTN

Les réseaux DTN (Delay Tolerant Network) désignent des réseaux capables de transmettre des informations debout en bout, même lorsque le réseau n'est pas connecté en permanence. Les DTN fonctionnent en mode déconnecté et ils accommodent la mobilité et l'énergie limitée des appareils de communication sans fil [46]. La figure 2.12 représente un exemple d'un réseau DTN.

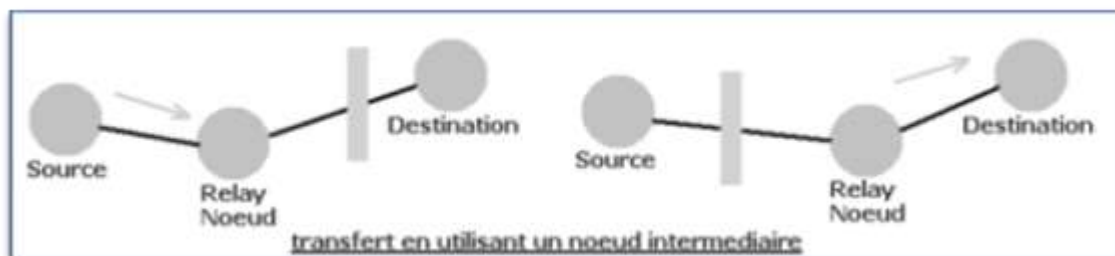


Figure 2.12 exemple d'un réseau DTN : En fonction de sa mobilité, soit le nœud relais est connecté à la source, soit il est connecté à la destination mais, généralement, pas aux deux simultanément ; il n'y a pas de connexion de bout en bout.

Notons que, les technologies sans fil DTN sont diverses, nous trouvons [46]:

- **La radio fréquences (RF)** : une forme de communication sans fil qui permet de transmettre l'information d'un terminal à une station de base, qui à son tour la transmet à un ordinateur hôte.
- **L'UWB (Ultra Wide Band)** : une technologie radio ultra large bande, utilisée pour la communication haut débit sur courte distance et avec une très faible puissance. La bande passante UWB est définie comme ayant une largeur d'au moins 500 MHz.

- **La liaison dégagée optique (Laser) :** une technologie qui a un avantage économique certain sur les solutions filaires. Où la seule condition pour l'installer est de garantir une vue dégagée des obstacles entre les deux points.

2.2.3.3. Caractéristiques des réseaux DTN

Les réseaux DTNs se distinguent des réseaux conventionnels par les caractéristiques suivantes [6]:

1. **Connectivité intermittente :** dans la plupart des cas, il est impossible d'avoir un chemin debout en bout. Un nœud se connecte à d'autres nœuds qu'occasionnellement et le lien est la ressource la plus rare dans le réseau.
2. **Le délai tolérable :** la latence de transmission de bout-en-bout est dominée par le délai dans les files d'attente. Les messages doivent être stockés dans la file d'attente des messages jusqu'à ce que le nœud rencontre un nœud voisin. Évidemment, la connexion dans un réseau DTN mènera à une longue période de latence afin que les applications doivent tolérer le retard de transmission large.
3. **La faible densité :** la densité de nœud est normalement beaucoup plus faible dans les réseaux DTNs par rapport aux réseaux traditionnels densément déployés.
4. **La mobilité des nœuds :** dans les réseaux DTNs les nœuds sont mobiles, par conséquent la topologie du réseau est changée fréquemment. En outre, la taille du tampon de capteurs est généralement limitée et les messages peuvent être stockés dans la file d'attente du tampon pour un temps assez long avant d'être envoyés, ce qui exige une bonne gestion de la file d'attente.

2.2.3.4. Fonctionnement des réseaux DTN

Le réseau DTN tient compte du cas où le réseau peut être partitionné. On considère dans ce cadre que si aucune route complète n'existe, à un instant donné, entre deux nœuds, on peut tout de même faire transiter un message dans le réseau de proche en proche entre les nœuds accessibles dans l'espoir que l'un d'entre eux sera à un moment donné à portée du destinataire. Cette technique consiste en général à utiliser des nœuds potentiellement mobiles (data mule) qui ne sont pas, a priori, les destinataires d'un message pour le mettre dans un cache local et le retransmettre si l'occasion se présente (voir Figure 2.13)[45].

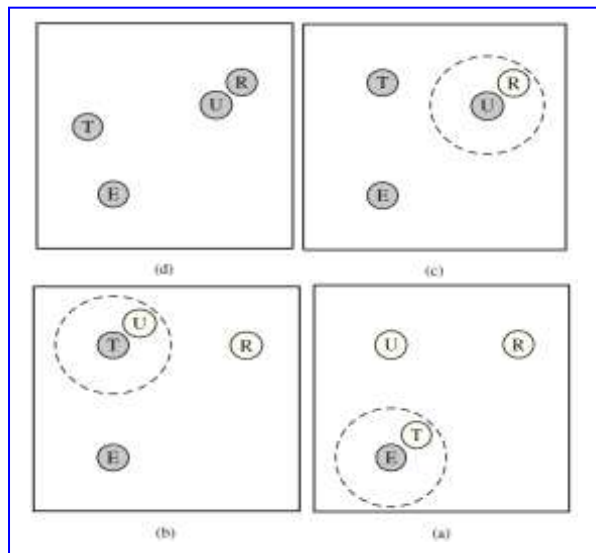
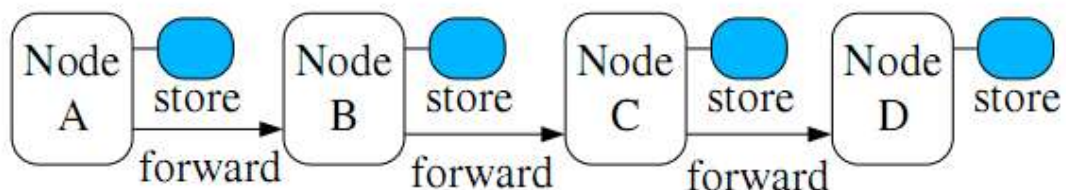


Figure 2.13 Principe de fonctionnement d'un réseau DTN : cette figure présente un scénario utilisant une approche opportuniste (DTN) pour transmettre un message du nœud E au nœud R en utilisant les nœuds T et U. Les nœuds porteurs du message sont grisés. La stratégie présentée dans cette figure consiste à transmettre systématiquement un message à tous les nœuds à portée.

Les DTNs surmontent les problèmes associés à la connectivité intermittente, aux délais longs ou variables, à la vitesse de transmission asymétrique et au taux d'erreur important en utilisant la technique de "**store and forward**" (stocker et retransmettre). Les méthodes de "store and forward" sont utilisées dans la vie quotidienne, par exemple : pour l'acheminement d'un courrier, d'un e-mail, d'un message vocal, etc. Pour cela, l'utilisation d'espace de stockage s'avère nécessaire



(Figure2.14) [61].

Figure 2.14 La technique Store and Forward: Les nœuds doivent mettre les paquets dans des tampons et les envoyer au prochain saut le plus approprié lorsque la connexion vers ce nœud devient disponible.

Les réseaux DTN mettent en œuvre les techniques «store and forward» de commutation de données en superposant au protocole existant une nouvelle couche protocolaire appelée *couche bundle (bundle layer)* (voir figure 2.15) dont le principal rôle est le stockage de données jusqu'à ce qu'une occasion de transmission se présente. La bundle layer permet de relier les spécificités des couches inférieures des régions différentes, ainsi une application peut communiquer à travers des réseaux de natures différentes [61]. Les paquets des données passant à travers la couche bundle sont regroupés en des unités basiques appelées bundles ou messages [6].

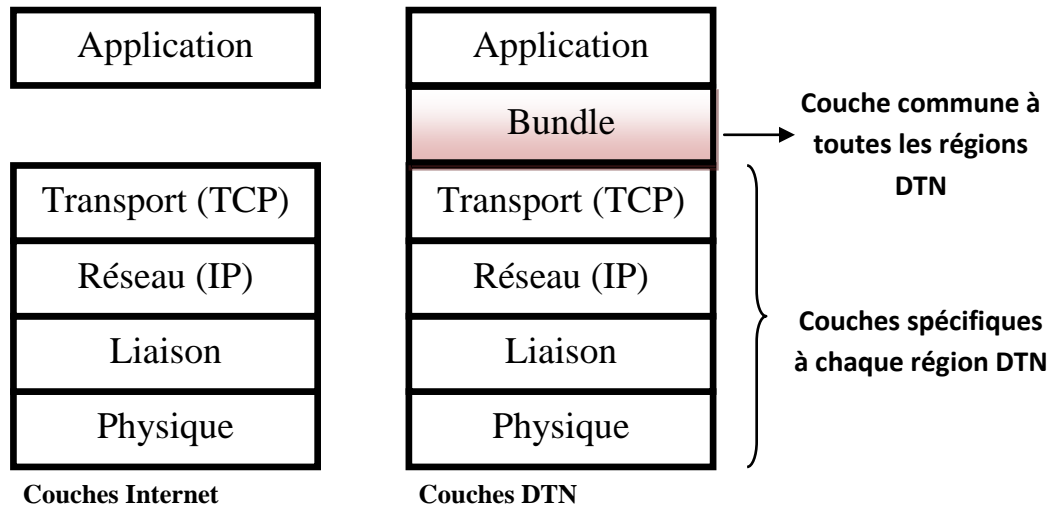


Fig2.15 Comparaison entre les couches Internet et les couches DTN : La couche bundle permet de relier les spécificités des couches inférieures des différentes régions, ainsi une application peut communiquer à travers des réseaux de natures différentes.

2.2.3.5. Routage dans les réseaux DTN

Un protocole de routage est utilisé pour trouver un chemin de livraison efficace en termes de temps à partir d'une source vers une destination. Dans un DTN, le délai de livraison d'un message sur un lien entre deux nœuds est déterminé par quatre éléments [6]:

- Temps d'attente de Connexion : temps d'attente pour que le lien soit connecté.
- Temps du buffer : Temps d'attente pendant lequel le message est sauvegardé dans la mémoire tampon.
- Temps de transmission: le temps de livrer le message de la mémoire tampon au lien.
- Temps de propagation: temps pour que le message traverse le lien.

Dans les réseaux DTN, les notions de chemin et de topologie, sont différentes puisque les contacts peuvent être intermittents et les nœuds sont mobiles. À cause de topologie variable dans le temps, les protocoles de routage traditionnels se révèlent inefficaces voire ne fonctionnent plus dans les DTN. Pour cela plusieurs protocoles de routages dédiés aux réseaux DTN ont été développés et regroupés selon plusieurs critères. Nous allons présenter dans ce qui suit quelques classifications des protocoles de routages dans ce type de réseau.

A. Les défis du Routage

L'un des principaux objectifs de conception de DTNs est d'échanger des données entre les nœuds en employant des liens opportunistes chez les nœuds pour la transmission. De toute évidence, la conception des protocoles de routage dans les DTNs est influencée par de nombreux défis. Dans ce qui suit, nous résumons quelques défis qui affectent le routage dans les DTNs [63] :

A1. Une connectivité intermittente

Le DTN est un réseau partiellement connecté en raison de la mobilité, le déploiement dispersé de nœuds et la mauvaise qualité de communication entre ces derniers. La connectivité du réseau varie dans le temps. Par conséquent, il est difficile à une connexion debout en bout entre le nœud source et le nœud destinataire ainsi que les techniques de routage dans le réseau conventionnel ne soient pas bien adaptées aux réseaux DTNs. Une connectivité intermittente signifie que les liens entre les nœuds sont opportunistes. Comment obtenir un lien opportuniste et transmettre un message est un problème difficile dans les DTNs.

A2. Une latence élevée

Les messages doivent être stockés dans la file d'attente des nœuds en raison de la connectivité intermittente, encourir plus de temps d'attente. Par ailleurs, le faible débit de données des DTNs présente plus de délai de transmission. La conception des protocoles de routage pour les DTNs devrait permettre de réduire le temps de latence de livraison le plus possible.

A3. Les ressources limitées

Les nœuds dans les DTNs sont équipés de modules de faible puissance, d'une taille mémoire tampon limitée, de batterie irremplaçable et d'une faible capacité de calcul, c'est-à-dire, les ressources des nœuds sont limitées. La limitation des ressources dégrade les performances des protocoles de routage.

A4. La taille du tampon

Le nœud attend habituellement une longue période de temps jusqu'à ce qu'il rencontre un autre nœud de sorte que les messages doivent être stockés dans sa file d'attente. Si la file d'attente est pleine, certains messages seront déposés, ce qui diminue le taux de livraison. Des stratégies de routage peuvent être nécessaires pour prendre en compte l'espace tampon limitée lors de prises des décisions de routage. En outre, il doit y avoir un système pour gérer l'espace mémoire.

A5. L'efficacité énergétique

Les nœuds dans les réseaux DTNs sont généralement alimentés par une batterie qui ne peut être remplacée facilement. Beaucoup d'énergie est consommée pour l'envoi, la réception et le calcul. Il existe un compromis entre la consommation d'énergie et la connectivité réseau. Le problème qui se pose c'est comment maintenir une connectivité acceptable tout en gardant l'énergie.

A6. Capacité de traitement

Les nœuds dans DTNs peuvent être très petits et ont une capacité de traitement faible, en termes de CPU et de mémoire. Ces nœuds ne seront pas capables d'exécuter des protocoles de routage complexes. Pour concevoir des protocoles de routage pour DTN, nous devons prendre en considération la puissance de calcul des nœuds.

B. Classifications des protocoles de routage dans les réseaux DTN

Une première classification (voir Figure 2.16) complète a été proposée par Jain et al. [65]. Celle-ci se base sur deux propriétés principales: la « réplication » où plusieurs copies du message sont utilisés et la « connaissance » qui indique comment obtenir des informations sur l'état du réseau et comment qu'une stratégie les utilise dans le but de prendre une décision de routage.

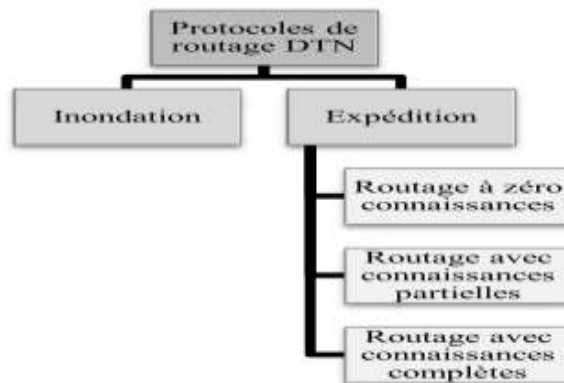


Figure2.16 Diagramme résumant la classification de Jain & Al[65]

Par la suite, des extensions de la première classification ont apparu (voir Figure2.17). Certaines ont continué à se baser sur les mêmes propriétés, mais en ajoutant d'autres protocoles de routage comme dans [66], d'autres ont formé de nouvelles classes afin de diminuer le taux de perte ou d'échec dans l'acheminement des paquets, le délai debout en bout ainsi que le nombre total de transmissions nécessaires à la transmission d'un paquet.

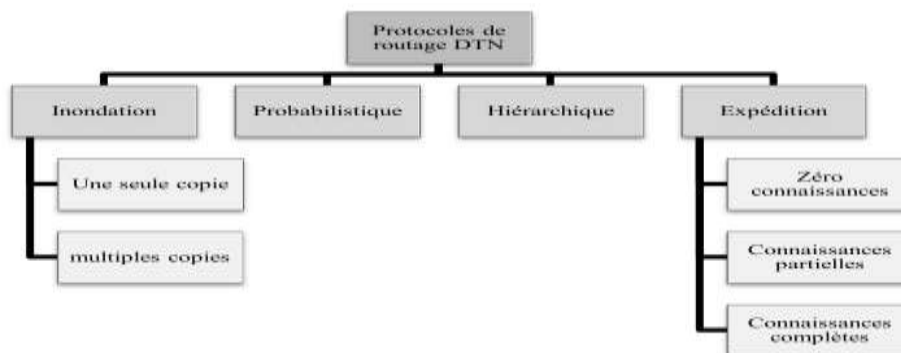


Figure2.17 Extension du diagramme de classification de Jain & Al

Parmi les classifications proposées, nous citons celle présentée par Benmessaoud[46], fondée sur les ressources consommées dont les deux métriques étudiées sont la « bande passante » et la « capacité de stockage ». La figure suivante montre les trois classes principales (voir Figure 2.18) :

1. La classe des protocoles non consommateurs de ressources.

Chapitre 2 : Réseaux sans fil

2. La classe des protocoles ne se souciant pas de la capacité des ressources et donc consomment sans limites.
3. La classe des protocoles tentant de réduire cette consommation.

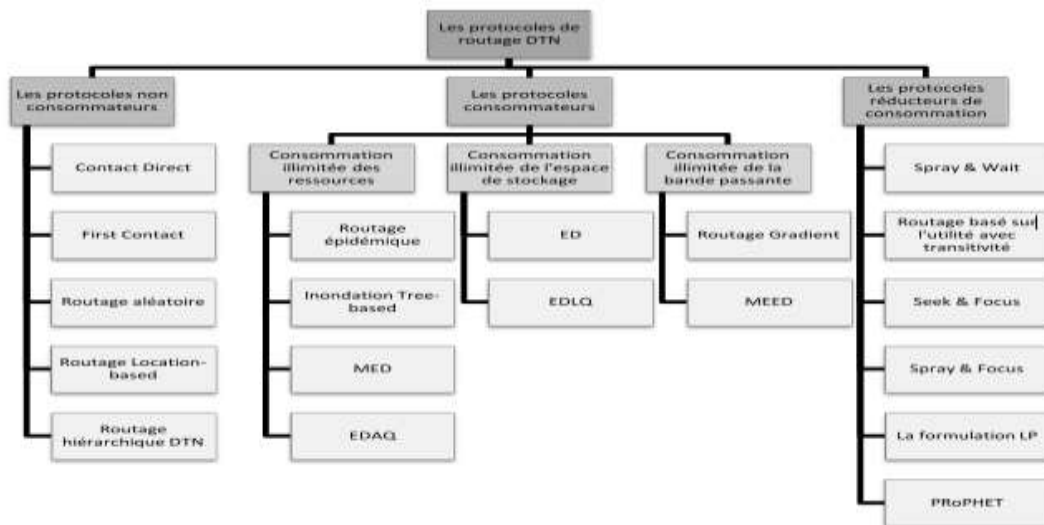


Figure 2.18 Classification des protocoles de routage DTN selon le taux de consommation de ressources

Gong et al. [63] ont présenté une nouvelle classification (voir Figure.2.19) composée de trois classes à savoir : Nonknowledge-Based Routing Protocols, Knowledge-Based Routing Protocols et Social-Based Routing Protocols. Les protocoles de la première classe sont fondés sur la réplication des copies de message sans aucune connaissance sur le réseau, tandis que les protocoles de la deuxième classe utilisent les connaissances sur l'environnement comme l'historique des contacts, le modèle de mobilité et la topologie du réseau pour optimiser les stratégies de routage et améliorer la performance du réseau. La différence entre cette nouvelle classification et l'ancienne réside dans la troisième classe « Social-Based Routing Protocols » dont le principe est de s'inspirer du comportement social (*Centralité, Amitié, communautaire, l'égoïsme*) pour résoudre les problèmes de routage. Un survol de ces protocoles a été présenté par ZHU et al. [67].

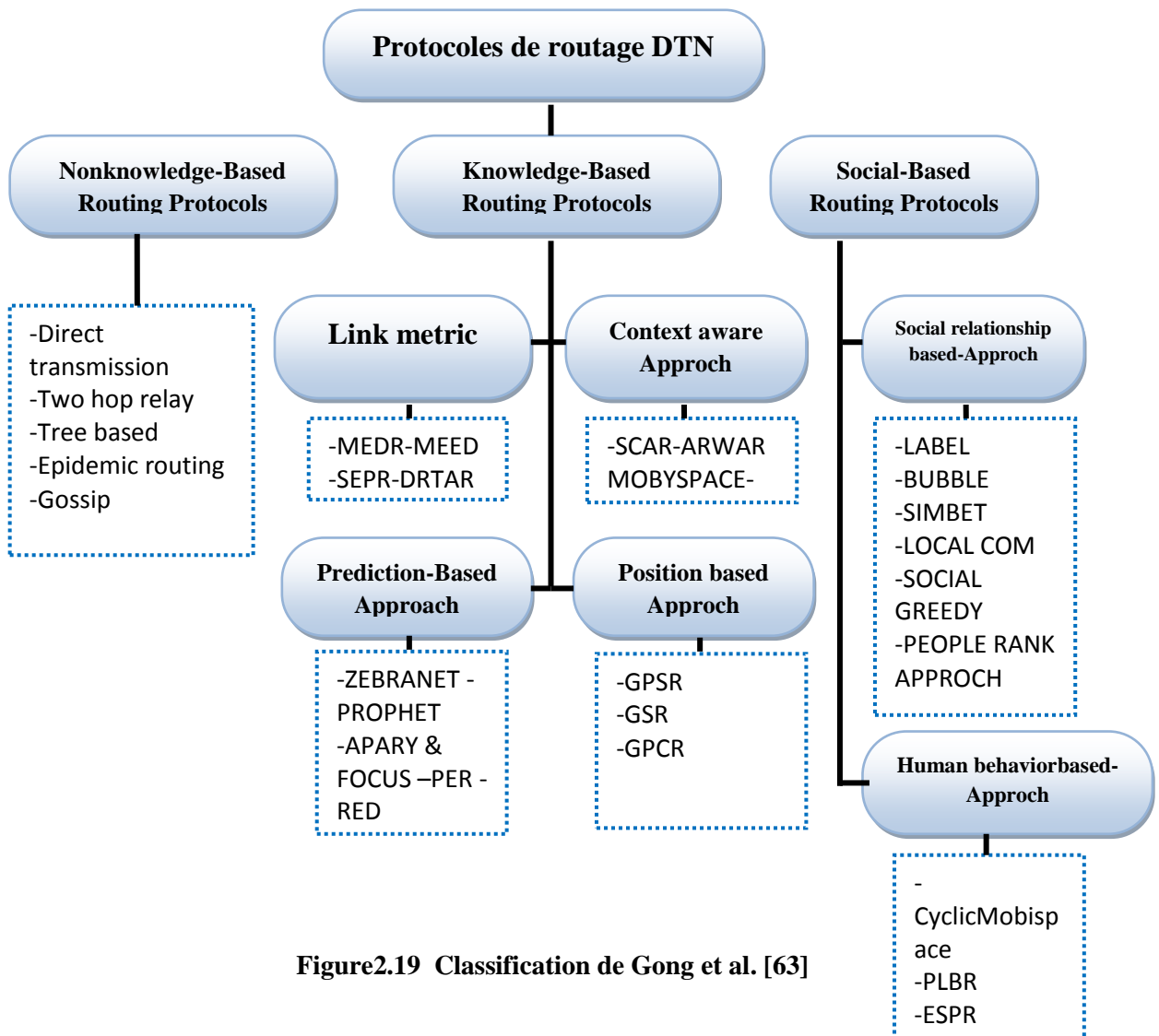


Figure 2.19 Classification de Gong et al. [63]

Afin d'obtenir une vue des différentes stratégies de routage DTN, MIN et al.[6] les ont classés en fonction de quatre critères: les copies de messages, le type d'information, le type de décision et le critère de décision (voir Figure.2.20).

1. Copies de messages : les protocoles de routage peuvent être classés en trois catégories, selon le nombre de copies du même message créé lors du routage à savoir : l'expédition, la réplique et les inondations (forwarding, replication, and flooding).

- Dans la catégorie **expédition**, une seule copie est expédiée par des nœuds successifs vers la destination. Seulement la source et la destination ont cette copie de message.
- Le principe de la **réplique** consiste à créer une copie pour chaque message pendant qu'il est transféré vers le nœud du prochain saut. Par conséquent, plusieurs nœuds intermédiaires ont chacun une copie du même message.

- L'**inondation** « **flooding** » est une réplification extrême telle que presque tous les nœuds du réseau ont une copie du même message.
 - 2. **Information de type** : pour soutenir des décisions de routage, chaque nœud dans le réseau maintient les informations de routage locales ou globales telles que des vecteurs de distance ou des états de lien. Les nœuds d'un seul saut et même de deux sauts dans le voisinage doivent échanger les informations de routage dans un plan local; cependant, les informations de routage sont propagées à tous les nœuds dans un plan global.
 - 3. **Décision type**: un système de nœud source fonctionne comme un protocole de routage source qui sélectionne un chemin de routage final au niveau du nœud source. Dans un système par saut, le chemin de routage peut être changé à chaque fois qu'il est transmis par un nœud intermédiaire.
 - 4. **Critère de décision**: le chemin de routage peut être vu comme des sélections successives des nœuds intermédiaires (next hop) de la source vers la destination. Pour chaque décision on a besoin d'évaluer si un nœud qui est rencontré est assez bon pour être un nœud next-hop. Différents facteurs d'évaluation ont été étudiés et peuvent être classés en trois catégories : propriétés du nœud, propriété de la liaison et propriété du chemin.
- Propriétés du nœud : nombre de nœuds en contact récent avec ce dernier et la taille de la mémoire tampon.
 - Propriété de liaison (Link) : l'historique des contacts, distance, sens de déplacement relatif.
 - Propriété du chemin : coût de livraison du chemin.

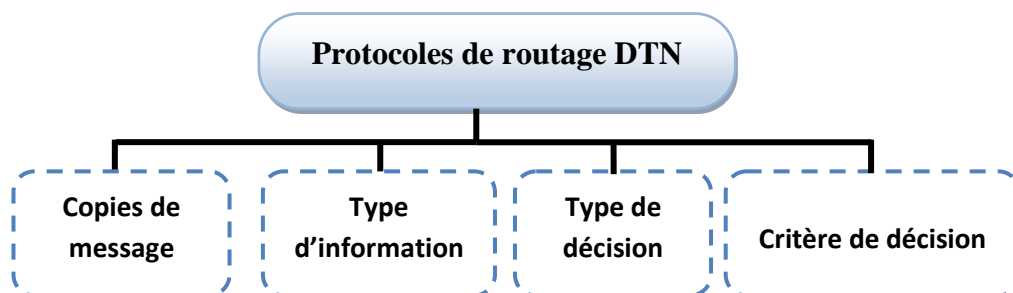


Figure 2.20 Classification de Min et al.[6]

C. Les routages les plus adoptés

C1. Routage Épidémique (basé sur l'inondation)

Les algorithmes épidémiques sont originellement proposés pour synchroniser les bases de données répliquées. Puis ces algorithmes ont été appliqués dans l'expédition de données dans un DTN. Par analogie, la liste des messages en attente de délivrance représente la base de données qui a besoin de synchronisation.[46][63][68] ont décrit le principe du routage épidémique comme suit : Chaque nœud a

Chapitre 2 : Réseaux sans fil

une file d'attente dans laquelle il stocke ses messages originaux et les messages des autres nœuds avec un identificateur unique pour chaque message. Lorsque deux nœuds se rencontrent, ils s'échangent un sommaire y compris la liste de tous les ID des messages qu'ils ont dans leurs tampons, puis ils s'échangent les messages qu'ils n'ont pas. Par conséquent, les copies de messages se propagent comme une «épidémie» dans le réseau en utilisant le maximum de ressources. Ce qui fournit une grande quantité de redondances, car tous les nœuds reçoivent tous les messages, mais qui rend cette stratégie extrêmement robuste. Le routage épidémique est relativement simple, car *il ne nécessite aucune connaissance sur le réseau*, pour cela il est classé dans la catégorie des protocoles basés sur l'inondation et non pas sur la connaissance. Pour cette raison, il a été proposé pour être utilisé comme repli si aucune autre meilleure méthode n'est disponible. L'inconvénient c'est qu'une énorme quantité de ressources est consommée, dû au grand nombre de copies, ce qui nécessite une grande capacité de stockage, de bande passante et d'énergie [46]. Ces raisons rendent ce type de routage impraticable dans les scénarios d'application les plus réelles dans lesquelles ces ressources ne sont pas suffisamment disponibles. Les nœuds dans un réseau DTN sont des appareils mobiles alimentés par des batteries de capacité de stockage et d'énergie limitée, ce qui exige la livraison ou la suppression de message lorsque la mémoire tampon est pleine. Le problème des ressources limitées a été traité par Elwhishiet al. Dans [68], ils présentent le problème de perte des messages lorsque la taille de ces derniers devient plus grande que la capacité du buffer. Elwhishiet al. [68] ont proposé un cadre de planification de message de tel sorte que le nœud prend la décision d'envoyer ou de supprimer le message, une telle décision est faite en évaluant l'impact de suppression des messages stockés dans les buffers sur le taux de livraison et le délai des messages. L'optimisation qui est faite dans ce type de routage consiste à associer à chaque message un nombre maximum de sauts (lorsque ce nombre atteint la valeur de 1, il ne peut plus être délivré qu'au destinataire). La détermination de cette valeur est importante, car plus ce nombre est grand et plus le message va se propager rapidement, mais aussi encombrer le réseau [45].

C2. Protocole PROPHET (basé sur la connaissance)

Prophète (Probabilistic Routing Protocol using History of Encounters and Transitivity) est un protocole de routage probabiliste qui utilise l'histoire de rencontres pour calculer la prédictibilité de livraison des nœuds qui indique la probabilité de rencontrer le nœud de destination. Chaque nœud maintient la prédictibilité de livraison de chaque autre nœud pour toutes les destinations connues. PROPHET utilise une stratégie très simple. Quand un nœud rencontre un autre nœud qui a une prédictibilité plus grande pour le message, il transmet ce message à ce nœud et garde encore ce message pour le transmettre à d'autres nœuds dans le futur. Le protocole PROPHET a un ratio de distribution supérieur à l'épidémie, avec une surcharge de communication beaucoup plus faible. Parmi les problèmes de ce protocole, on retrouve le suivant : lorsque le nœud original du message a une très faible

prédictibilité pour la destination, il va transmettre ce message vers tous les nœuds qu'il rencontre, jusqu'à ce que ce message soit supprimé de la file d'attente (souvent, le nœud original garde ses messages pendant longtemps). Différentes stratégies ont été utilisées pour gérer la file d'attente des nœuds comme *FIFO*, *DOA* (le principe est de supprimer en priorité le message qui est le plus vieux) et *DLE* (supprimer le message qui a la prédictibilité la plus petite parce qu'il y a peu de chance de transmettre ce message à la destination). Lindgren et al. [69] ont proposé un protocole *PROPHET* et l'ont comparé avec le protocole de routage épidémique à travers les simulations et selon les métriques suivantes ; la capacité de livraison des messages, le délai de livraison d'un message et le nombre de messages échangés. Ils ont montré que *PROPHET* est capable de délivrer plus de messages que le routage épidémique avec une surcharge de communication plus basse. Une métrique probabiliste appelée prédictibilité de livraison « *delivery predictability* » ainsi que la stratégie *FIFO* pour la gestion de mémoire ont été utilisées.

- **Calculs de prédictibilité**

La prédictibilité est la probabilité qu'un message arrive à la destination via un nœud intermédiaire donné. Chaque nœud possède une liste de toutes les destinations (éventuellement tous les nœuds) avec les prédictibilités associées. Il existe plusieurs techniques qui peuvent être utilisées pour calculer ces valeurs afin de décider quels nœuds sont les meilleurs candidats pour porter un message particulier. Mascolo et al.[70] ont utilisé *la théorie des filtres de Kalman* pour réaliser une prédiction plus réaliste de l'évolution du contexte d'un nœud et d'optimiser l'utilisation de la bande passante. Comme il est indiqué dans ce travail, l'échange des tables de routage qui stockent les probabilités de livraison est un processus potentiellement coûteux et donc inutilement lorsque ces informations sont assez facilement prévisibles, ce qui montre l'utilité des filtres de Kalman. Il est à noter que l'un des principaux avantages du filtre de Kalman est qu'il ne nécessite pas le stockage de toute l'histoire passée du système, ce qui convient pour un cadre mobile dans lequel les ressources mémoire peuvent être potentiellement très limitées. Lindgren et al [69] ont présenté le calcul des prédictibilités de livraison en trois parties. La première chose à faire est de mettre à jour la métrique à chaque fois qu'un nœud est rencontré, de sorte que les nœuds qui sont souvent rencontrés ont une prédictibilité de livraison élevée. Ce calcul est illustré dans l'équation 1, où P_{init} est une constante, il s'agit de la prédictibilité initiale prise par tous les nœuds qui se rencontrent pour la première fois et qui dépend a priori de la mobilité des nœuds, $P_{(a,b)}$ est la probabilité qu'un nœud a rencontre un nœud b .

$$P_{(a,b)} = P_{(a,b)old} + (1 - P_{(a,b)old}) \times P_{init} \quad \text{Eq1}$$

Deuxièmes, si des paires de nœuds ne rencontrent pas les uns avec les autres pendant un certain temps, ils sont moins susceptibles d'être de bons expéditeurs de messages donc, les valeurs de prévisibilité de livraison doivent être décrémentées. Ce renouvellement est illustré par l'équation (2), où γ ($0 < \gamma < 1$) est une constante qui

définit la valeur dont doit être décrétementée la prédictibilité par unité de temps. En pratique cette valeur dépend de la mobilité et de la densité des nœuds. k est la durée depuis la dernière mise à jour de $P(a,b)$.

$$P_{(a,b)} = P_{(a,b)old} \times \gamma^k \quad \text{Eq2}$$

Troisièmement, La prévisibilité de livraison a également une propriété transitive, l'équation (3) montre comment cette transitivité affecte la prévisibilité de livraison, où β ($0 < \beta < 1$) est une constante reflétant l'influence de la transitivité.

$$P_{(a,c)} = P_{(a,c)old} + (1 - P_{(a,c)old}) \times P_{(a,b)} \times P_{(b,c)} \times \beta \quad \text{Eq3}$$

Quand les nœuds a et b se rencontrent, le nœud utilise la formule (1) et la formule (3) pour renouveler les valeurs de prédictibilité. Selon (1), a affecte à ces nœuds une prédictibilité de plus en plus grande aux nœuds qu'il rencontre souvent. Selon (3), si a rencontre b très fréquemment et si b a une très grande prédictibilité pour c (un autre nœud), a obtient aussi une très grande prédictibilité pour c . La formule (2) se calcule pour chaque période k de temps, ce qui contribue à diminuer la prédictibilité des nœuds rencontrés peu fréquemment par a .

Burgess et al. [71] ont utilisé une autre méthode de prédictibilité pour le protocole MaxProp. Quand le nœud A rencontre le nœud B , A augmente sa prédictibilité pour B de $k=1$ unités, après A re-normalise toutes ses prédictibilités pour obtenir une somme de toutes ses prédictibilités égale à 1. Prenons un exemple dans lequel le réseau comporte 4 nœuds. Au départ, A a les prédictibilités suivantes pour les autres nœuds :

$$P(A,B)=P(A,C)=P(A,D)=0.25.$$

$$\text{Quand } A \text{ rencontre } C, \text{ il modifie } P(A,C)=0.25+k=1.25 / (k=1),$$

$$\text{Après la re-normalisation, } P(A,B)= P(A,D) =0.125 \text{ et } P(A,C)= 0.625$$

Dans cette méthode, quand un nœud rencontre un autre pour lequel il a une prédictibilité faible, une telle modification de la prédictibilité est très importante. Elle donne une très grande valeur tout de suite au nouveau nœud rencontré juste après la re-normalisation. Donc un k plus petit semble plus approprié.

2.2.3.6. Domaines d'application de réseaux DTN

De nombreux domaines d'application utilisant les réseaux DTNs ont été étudiés depuis 2004, y compris la communication dans l'espace cosmique, suivi de la faune, la surveillance sous-marine, les réseaux sociaux, champs de bataille, les régions en développement et réseaux ad hoc véhiculaires (VANet). Par exemple, l'IPN (Interplanetary Internet : internet interplanétaire) est un projet qui étudie la communication dans l'espace cosmique. Le ZEBRANET est un projet qui utilise les capteurs afin d'étudier l'interaction entre les zèbres. Le DAKNET un projet en 2004 offre une communication pour les régions éloignées en Inde. Le projet HAGGLE utilise des appareils portatifs pour échanger des informations entre les personnes. Le

projet CONDOR soutenu par « US Marine Corps » étudie la communication dans les champs de bataille. Le projet UMASDIESELNET examine la transmission de données entre les autobus se déplaçant autour du campus de l'université de Massachusetts [6].

2.2.4. Le réseau Ad hoc et le réseau DTN

Un réseau tolérant aux délais est une évolution des réseaux mobiles ad hoc « MANET ». C'est un réseau de faible densité et de connectivité intermittente où la communication bout en bout n'est pas toujours disponible pour la transmission des messages [66]. Cependant, la probabilité de transmission du message est élevée dans les DTNs par rapport aux MANET grâce à l'approche store and forward. Dans les MANETs, un protocole de routage doit établir un chemin de bout en bout de la source à la destination en utilisant les nœuds intermédiaires. Cette connectivité devrait être maintenue jusqu'à ce que la transmission du message soit terminée. Ce n'est pas le cas dans les DTNs, où le chemin de bout-en-bout est très rare et instable dans la nature et la possibilité d'établir un chemin complet est négligeable. Seules les applications à forte latence sont utilisées dans un DTN où la latence peut être de l'ordre de quelques heures, voire même de quelques jours [72].

La déconnexion va se passer plus souvent dans les réseaux mobiles ad hoc avec un modèle de mobilité en groupe. Cependant, la plupart des protocoles de routage DTN actuels sont conçus sur le modèle de la mobilité individuel et non pas sur le modèle de mobilité en groupe. Pour cela, XIE et al [73] ont proposé un nouveau protocole de routage appelé « groupe epidemic routing » (G-ER), qui se base sur le protocole de routage « épidémique » de réseaux DTN. Le principe de ce nouveau protocole est de considérer le groupe comme un seul nœud pour pouvoir l'adapter au modèle de mobilité en groupe.

Les approches DTN sont très bien adaptées aux caractéristiques des réseaux MANets, car elles utilisent les liens disponibles pour transmettre un message au lieu de supposer l'existence d'une route. Cependant, suivant les contextes de mobilités dans lesquels on se trouve, on peut ressentir le désir d'adapter la méthode de diffusion [45]. Par exemple, *OTT et al* [74] proposent une solution hybride qui laisse à l'application le choix d'utiliser AODV ou un protocole DTN. Mascolo et al. [70] ont présenté l'algorithme (CAR) Context-AwareRouting qui est une nouvelle approche pour la communication asynchrone dans les réseaux mobiles ad hoc partiellement connectés. Le choix de protocole DSDV ou de protocole basé sur la prédictibilité dépend de l'appartenance du nœud destinataire au même cloud de la source. La prédiction des attributs de l'information du contexte se fait avec les filtres de Kalman. *MASCOLO et al* ont évalué l'algorithme CAR en utilisant le simulateur OmNet++. Un autre algorithme a été conçu par le même groupe de recherche, Musolesi et al. [75] ont conçu l'algorithme SCARA Context-aware Adaptive Routing dans les réseaux de capteurs tolérant aux délais « DTN Delay Tolerant Mobile Sensor Networks ». Le projet *SARAH* (Services distribués Asynchrones pour Réseaux mobiles Ad Hoc) a été

Chapitre 2 : Réseaux sans fil

financé de 2005 à 2009 par l'Agence Nationale de la Recherche dans le cadre du programme ARA SSIA 2005 (Action de Recherche Amont–Sécurité, Systèmes embarqués et Intelligence Ambiante). Il avait pour objectif d'étudier les problèmes posés par les réseaux mobiles ad hoc (ou MANET) discontinus et leur aptitude à supporter l'acheminement de données et le déploiement de services distribués.

2.3. Conclusion

Ce chapitre est consacré à la présentation des réseaux sans fils, plus précisément les réseaux MANet et les DTNs. Premièrement, un survol des réseaux MANET a été présenté dans lequel nous avons introduit les MANETs et leur pile protocolaire et expliqué les trois classes des protocoles de routage à savoir les protocoles proactifs, les protocoles réactifs et une classe des protocoles hybrides. Par ailleurs, nous avons détaillé pour chaque classe un protocole de routage. Il s'agit respectivement des protocoles AODV, OLDR et ZRP. Finalement, nous avons cité quelques domaines d'application des réseaux MANETs.

Aussi, nous avons expliqué la notion des réseaux DTNs, montré le contexte de leur apparition et expliqué leur fonctionnement selon le concept du Store and Forward. De plus, nous avons détaillé l'architecture de ce type de réseaux qui est caractérisée par la présence d'une couche supplémentaire appelée : « Bundle » située au-dessus de la couche transport. Après, nous avons présenté les classifications les plus importantes des protocoles de routage proposées pour les environnements intermittents. Il en existe bien évidemment d'autres qui sont, soit des dérivés de ceux qui ont été cités ci-dessus (dans le sens améliorations), soit contenant des protocoles adaptés à des situations particulières, qui ne sont pas généralement très connus. Beaucoup de projets et travaux ont été réalisés à base de cette nouvelle technologie, c'est ce que nous avons résumé dans la section : « domaines d'application ».

3

CHAPITRE

Méthodes d'optimisation

(Méta-heuristique PSO)

Chapitre 3 : Méthodes d'optimisation

3.1.Introduction

La résolution d'un problème d'optimisation consiste à utiliser un espace de recherche et à tenter de maximiser ou minimiser une fonction objective donnée. Les méthodes de résolution des problèmes d'optimisation sont principalement classées en deux catégories : les méthodes exactes (complètes) et les méthodes approchées (incomplètes). Les méta-heuristiques appelée aussi méthodes de recherche globale font partie des méthodes approchées. Ces méthodes visant à résoudre des problèmes d'optimisation difficiles pour lesquels soit les méthodes exactes existantes ne fournissent pas de résultats optimaux soit aucune méthode exacte n'est, à ce jour découverte. Un problème d'optimisation est présenté par [76] comme étant une boîte noire pourvue d'un certain nombre d'actionneurs et dont la sortie est une valeur réelle. Les variables du problème représentent les actionneurs auxquels il est possible d'attribuer un certain nombre de valeurs. Le problème d'optimisation peut posséder plusieurs valeurs comme sortie. L'optimisation a pour but de trouver les valeurs des variables qui produisent la meilleure valeur de sortie possible (maximiser ou minimiser).

3.2.Classification des méthodes d'optimisation

Bien que les problèmes d'optimisation combinatoire soient souvent faciles à définir, ils sont généralement difficiles à résoudre. En effet, la plupart de ces problèmes appartiennent à la classe des problèmes NP-difficiles et ne possèdent donc pas à ce jour de solution algorithmique efficace valable pour toutes tailles du problème traité. Un problème appartient à la classe de problèmes de décision polynomial (P) s'il peut être résolu par un algorithme a de complexité $O(N^k)$ où k est une constante et N la taille du problème. Par contre, un problème fait partie à la classe de problèmes de décision non polynomiale (NP) s'il peut être résolu en temps polynomial par un algorithme non déterministe. Généralement, les méthodes de résolution des problèmes d'optimisation peuvent être classées en deux classes différentes pour la recherche d'une solution: les méthodes exactes et celles approchées.

Nous nous intéressons en particulier à l'étude des algorithmes basé sur le comportement d'essaim qui sont des méta-heuristiques manipulant des populations de solutions et plus spécifiquement l'optimisation par Essaims Particulaires (OEP) ou PSO (Particle Swarm Optimization). Les méthodes que nous aborderons dans la suite de ce mémoire apparaissent en couleur grise sur la figure 3.1.

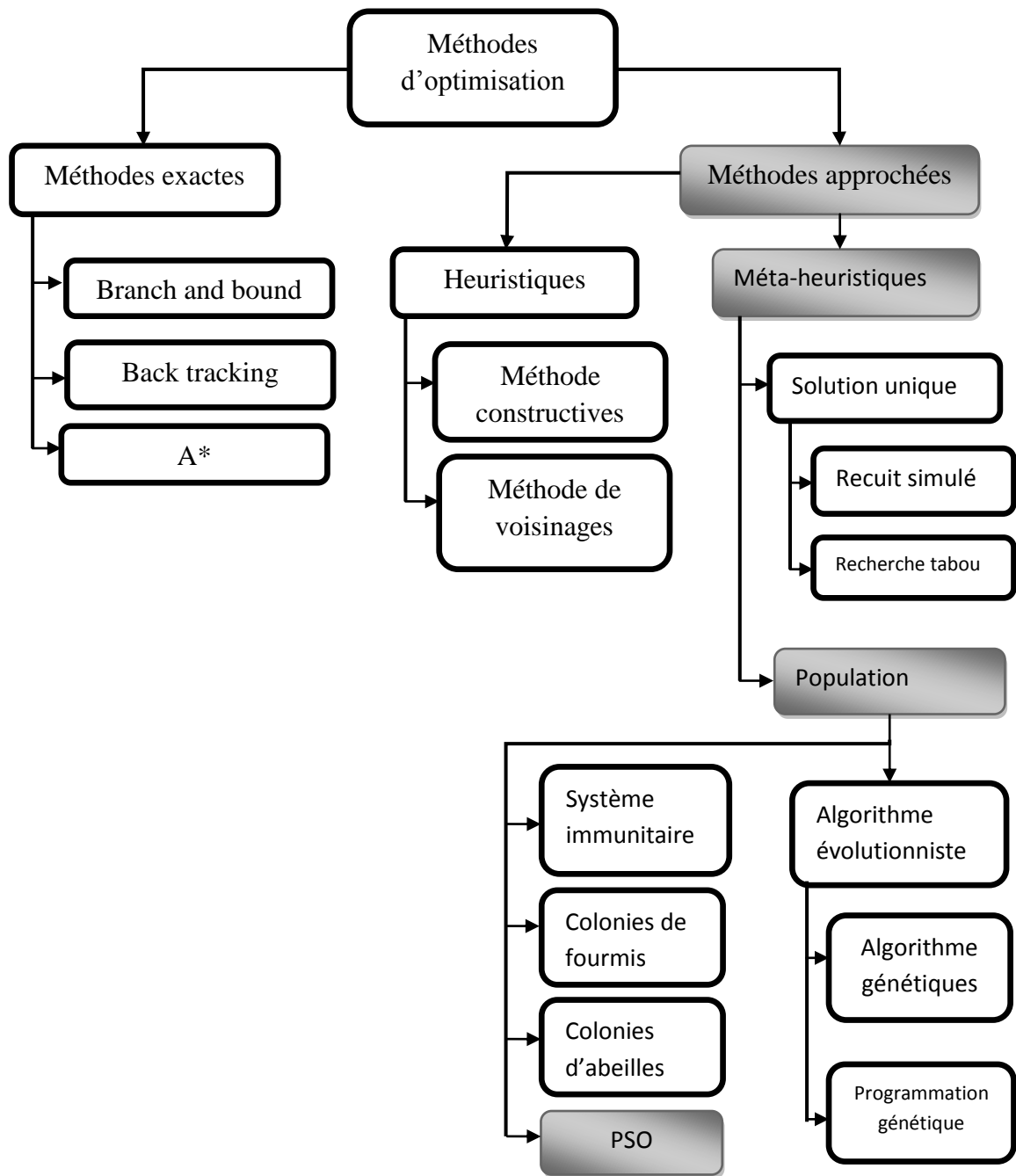


Figure 3.1 Classification des méthodes d'optimisation

3.2.1. Méthodes exactes (déterministes)

Cette catégorie travaille sur les problèmes de taille raisonnable. Le temps de calcul nécessaire d'une telle méthode augmente en général exponentiellement avec la taille du problème à résoudre. Elle repose sur l'exploitation d'un algorithme qui donne une solution optimale et l'énumération de l'ensemble des solutions de l'espace de recherche. Parmi ces méthodes on peut citer:

- **Branch and Bound** : Cette méthode repose sur deux procédures ; la séparation et l'évaluation. Un arbre de recherche représente toutes les solutions possibles est construit. Les branches de l'arbre sont évaluées et

Chapitre 3 : Méthodes d'optimisation

comparées avec une valeur seuil du critère à optimiser afin d'éviter les branches inutiles (contenant des solutions non intéressantes ou non réalisables). Comme toutes les méthodes de cette catégorie, la méthode de la recherche arborescente (Branch and Bound) est utile avec les problèmes de petites tailles et génère des problèmes face aux applications de taille importante [77].

3.2.2. Méthodes approchées (non déterministes)

Ces méthodes cherchent à trouver une solution de bonne qualité (acceptable) en un temps de calcul raisonnable sans garantir pour autant l'optimalité de la solution obtenue. Deux grandes classes sont distinguées dans cette catégorie.

3.2.2.1. Les heuristiques

Les méthodes qui appartiennent à cette classe se basent sur l'utilisation des informations collectées précédemment pour choisir la nouvelle solution à évaluer. Contrairement aux méthodes exactes, ces méthodes peuvent résoudre les problèmes de grandes tailles dans un temps raisonnable. Les méthodes constructives et celles de voisinages sont deux sous classes des heuristiques.

3.2.2.2. Les méta-heuristiques

Les méta-heuristiques consistent fondamentalement à faire évoluer une configuration de référence (configuration courante) en la remplaçant itérativement par une nouvelle configuration choisie dans son voisinage. Cependant, contrairement aux heuristiques, l'algorithme et l'emploi d'une méta-heuristique sont complètement indépendants du problème à traiter [77]. On peut partager les méta-heuristiques en deux grandes classes: les méta-heuristiques à solution unique et celles à solutions multiples ou population de solution. Dans les méta-heuristiques à solution unique on cite la méthode de recuit simulé [78] et la recherche Tabou [79][80]. Il existe une famille de méta-heuristiques qui consiste à faire évoluer un ensemble de solutions (population) : ces techniques sont appelées les algorithmes évolutionnistes. Les algorithmes génétiques et la programmation génétique font partie de ces algorithmes évolutionnistes. Un algorithme évolutionnaire nécessite une représentation des individus qui sont les solutions du problème qu'on cherche à résoudre. La représentation est basée sur le codage binaire de l'information [81]. Le principe général de la méthode est présenté dans l'algorithme suivant :

1. Initialisation de la population P_i : remplir aléatoirement la chaîne chromosomique de chaque individu.
2. Évaluer chaque individu de la population courante.
3. Créer la génération suivante en sélectionnant statiquement les géniteurs.
4. Réaliser les croisements sur les chaînes chromosomiques.
5. Réaliser les mutations.
6. Retourner en 2

Figure 3.2 Pseudo code des algorithmes évolutionnistes

3.2.2.2.1. Les algorithmes génétiques

Les algorithmes génétiques sont des algorithmes d'optimisation qui reposent sur des techniques dérivées de la génétique et de l'évolution naturelle : croisements, mutations, sélection, etc. Les algorithmes génétiques ont déjà une histoire relativement ancienne, puisque les premiers travaux de John Holland sur les systèmes adaptatifs remontent en 1962[82].

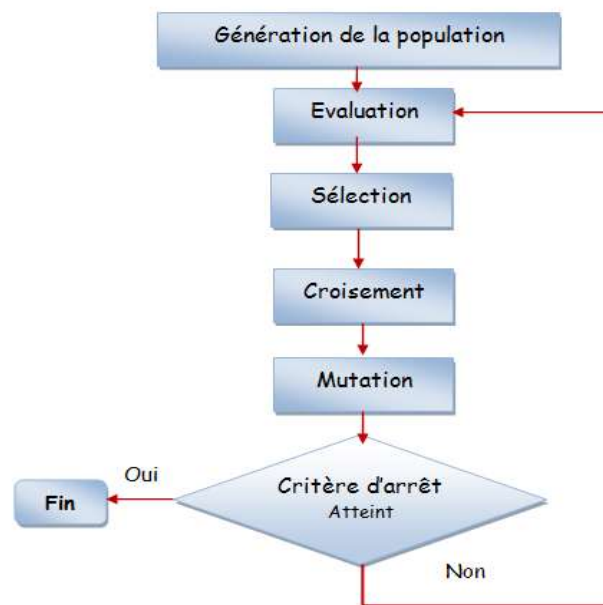


Figure 3.3 Principe de l'AG : En partant d'une population initiale générée aléatoirement, Tant qu'un critère d'arrêt n'est pas atteint, on applique l'opérateur de sélection pour élire les individus candidats pour se reproduire. Ensuite, on applique sur ces derniers les opérateurs de croisement et de mutation pour créer de nouveaux individus.

3.2.2.2.2. La programmation génétique

La programmation génétique proposée par JOHN R. KOZA est une variante des algorithmes génétiques. Sa particularité est que les solutions sont codées sous la forme d'arbres. La programmation génétique a été appliquée à des programmes en LISP qui est un langage utilisé en intelligence artificielle [81].

3.2.2.2.3. Optimisation par colonies de fourmis

ACO (Ant colony optimization) est une méta-heuristique utilisée pour trouver des solutions approximatives à des problèmes d'optimisation difficiles introduit par Clorniet al. [83]. Le principe est d'utiliser des fourmis artificielles qui s'inspirent du comportement des fourmis que l'on peut observer dans la nature qui consiste à l'exploitation des ressources alimentaires chez les fourmis et l'utilisation de phéromones comme un moyen de communication.

3.2.2.2.4. Optimisation par essaim particuliers

1. Principe

L'optimisation par essaim de particules (PSO) développée par Eberhart et Kennedy en 1995 [84] est une technique d'optimisation basée sur une population de solutions; il a inspiré du comportement social des interactions dans l'essaim d'oiseaux

Chapitre 3 : Méthodes d'optimisation

ou les bancs de poissons qui tentent d'atteindre une destination inconnue. PSO est une collection de particules ou (agents) dispersés dans un espace à N dimensions et qui cherchent la meilleure (optimum) solution en mettant à jour les générations. Par rapport à d'autres heuristiques, PSO présente quelques particularités intéressantes, comme l'efficacité est due à la collaboration plutôt qu'à la compétition. Chaque particule i est caractérisée par les informations suivantes :

- Sa position x_i qui représente la solution au problème.
- Sa vitesse v_i
- La meilleure position personnelle **pbest**
- La meilleure position globale «**gbest**» trouvée par les particules voisins.

Le degré d'optimalité est mesuré par une fonction fitness (aptitude) définie par l'utilisateur. Chaque particule utilise donc l'information dont il dispose par rapport à sa propre histoire et à celle de ses voisins afin de déterminer la direction qu'il doit prendre ainsi que sa vitesse de déplacement.

Les vitesses et les positions sont mises à jour selon les deux équations suivantes [84] [85]:

$$V_i^{K+1} = w.V_i^K + C_1.rand_1.(pbest_i^k - x_i^k) + C_2.rand_2.(gbest_i^k - x_i^k) \quad \text{Eq4}$$

$$X_i^{K+1} = X_i^K + V_i^{K+1} \quad \text{Eq5}$$

Tel que : K est le nombre d'itération, w est le coefficient d'inertie, C_1 et C_2 sont les coefficients d'accélération, $rand_1$ et $rand_2$ sont des nombres aléatoires distribués dans l'intervalle $[0,1]$.

La partie de l'apprentissage cognitive, $C_1.rand_1.(pbest_i^k - x_i^k)$ représente la tendance des particules à retourner à la meilleure position qu'il a visité à ce jour dans l'équation de vitesse. Tandis que la partie de l'apprentissage sociale, $C_2.rand_2.(gbest_i^k - x_i^k)$, représente la tendance des particules à être attirés vers la position de la meilleure position trouvée par l'essaim [86] (voir figure 3.4).

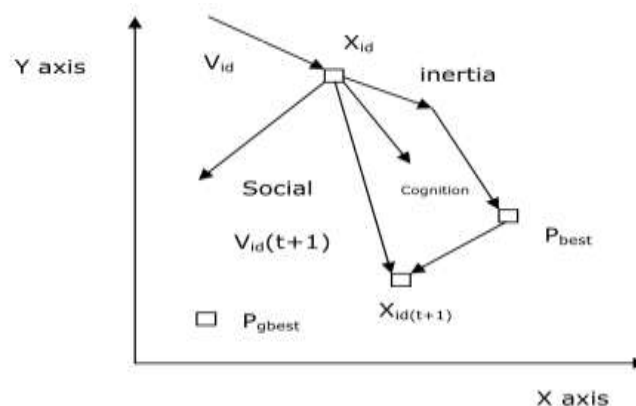


Figure 3.4 Schéma de principe de recherche de PSO

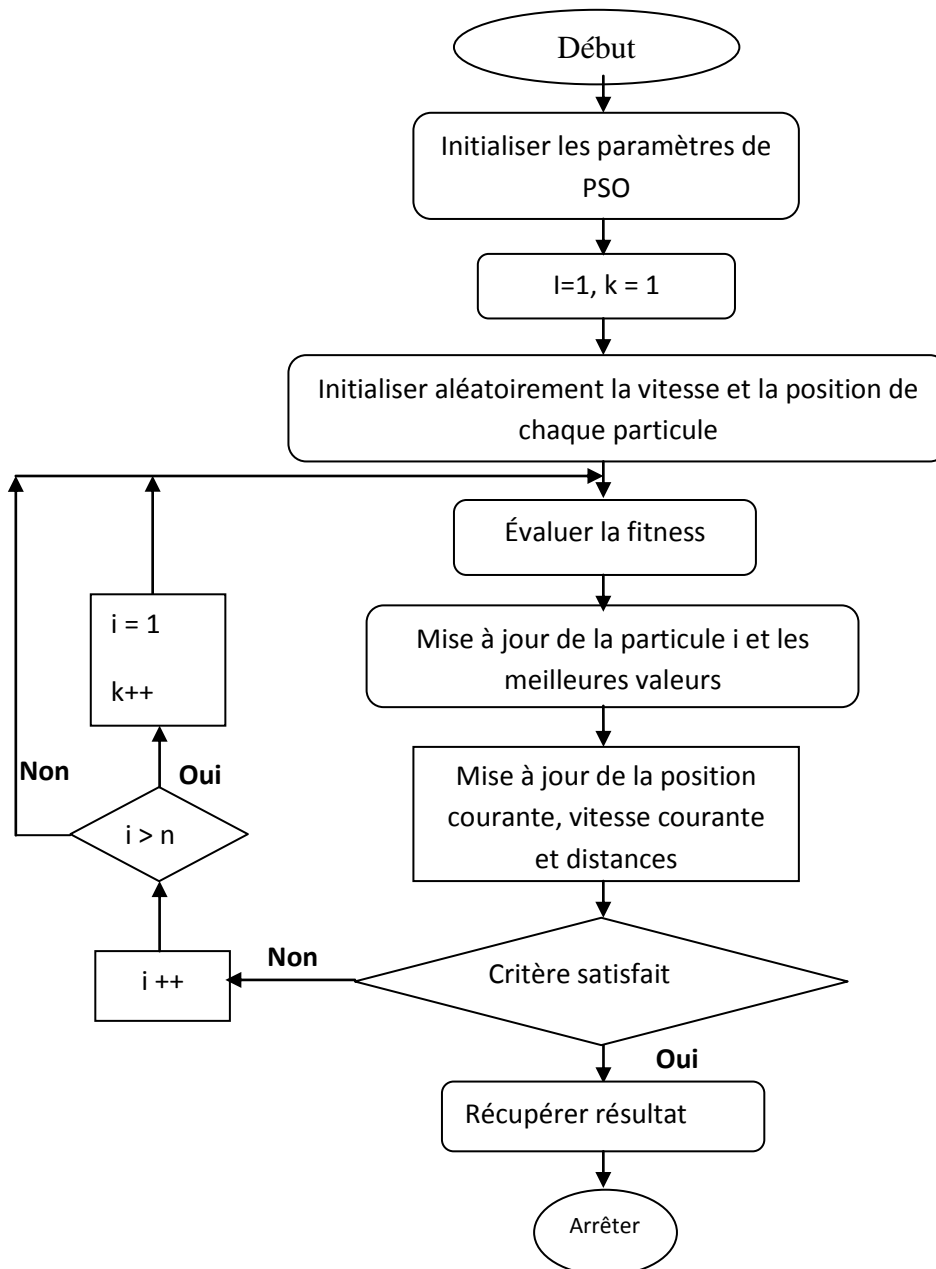


Figure 3.5 Principe de PSO

Le PSO, est un outil d'optimisation basé sur une population d'individus. Il a été prouvé que PSO peut donner de bons résultats pour les problèmes d'optimisation combinatoires discrets [87] et continus avec une fonction mono-objective ou multi-objectives [88]. La méthode PSO est favorablement comparable aux autres méthodes d'optimisation, comme les algorithmes génétiques et les algorithmes d'abeilles [82].

2. Différents types de PSO

Trois catégories principales de PSO sont distinguées par Blackwell [89][90][91] à savoir:

Chapitre 3 : Méthodes d'optimisation

1. Les essais biologiques (Bio-Swarm) : ils sont utilisés pour le développement des modèles scientifiques des systèmes naturels.
2. Les essais de simulation (Simulation Swarm): Comme le modèle de Reynolds et celui de Heppner.
3. Les essais sociales (Social Swarm) : les informations provenant d'un réseau topologique sont utilisées pour définir la manière d'interaction entre individus, comme la méthode PSO.

3. Applications des algorithmes de PSO

L'utilisation de techniques d'optimisation dans les problèmes de routage et d'exploration est un sujet très important pour les chercheurs. PSO a été largement appliquée dans les fonctions d'optimisation et précisément pour résoudre les problèmes de routage de réseau [92][93]. Nair et al [94] présentent une nouvelle idée qui permet aux nœuds de prendre des décisions et d'effectuer un routage efficace des paquets. Les résultats montrent que l'approche de routage avec PSO a atteint le chemin optimal de nœuds. Dans ce cas, les particules représentent les multiples trajets (entre deux nœuds) et que l'objectif est de trouver la particule qui a le maximum de bande passante associée au lien de la particule. Li et al [95] présentent un algorithme PSO adaptatif pour le problème de routage anycast (APSO-AR) pour résoudre le problème de routage anycast (ARP) avec de multiples contraintes de QoS. L'objectif de (APSO-AR), c'est de choisir un chemin praticable qui peut répondre à la condition de la bande passante, du délai et du taux de perte de paquet et son coût est minimum dans tous les chemins à partir du nœud source vers un autre nœud du réseau. Zhuo et al. [96] ont transformé le problème de routage multicast (QoS-MRP) qui est un problème NP-complet en un problème multi-objectif d'optimisation combinatoire. Un PSO discret efficace est proposé pour trouver un bon compromis entre trois paramètres: délai maximum, le coût de l'arbre et le délai moyen d'attente. Kukururu et al. [86] ont utilisé une approche fondée sur PSO pour résoudre le problème de la couverture du réseau de capteur mobile. L'objectif principal est de réduire la distance entre les nœuds voisins et la maximisation de couverture dans le réseau. D'autres travaux sont illustrés dans le tableau suivant [97]:

Problème	Auteurs	Référence
Problème de voyageur de commerce	Onwubolu and Clerc	[98]
Assignation des tâches	Salman, Imtiaz and Al-Madani	[99]
Réseaux de neurones	Kennedy, Eberhart, and Shi Settles, Rodebaugh and Soule	[100] [101]
Bioinformatiques	Correa, Freitas and Johnson Georgiou,	[102]

Chapitre 3 : Méthodes d'optimisation

Systemes électriques	Venayagamoorthy	[104]
Analyse de la classification	Chen and Ye	[105]
Problèmes Dynamiques	Carlisle and Dozier Eberhart and Shi	[106] [107]
Problèmes Multi-objectifs	Moore and Chapman Fieldsend and Singh Hu and Eberhart Parsopoulos and Vrahatis	[108] [109] [110] [111]

Tab3.1 Applications des algorithmes de PSO

3.3. Avantages et inconvénients des méthodes approchées

Les méta-heuristiques présentent en même temps de nombreux avantages, mais aussi un certain nombre de limitations. Les différents avantages ainsi que les inconvénients de ces méthodes sont résumés ci-dessous [112]:

- ⊕ Généralité et application possible à une large classe de problèmes,
- ⊕ Efficacité pour de nombreux problèmes,
- ⊕ Fournir des solutions approchées de bonne qualité pour un grand nombre de problèmes d'optimisation classiques d'applications réelles de grandes tailles.

- ⊗ Possibilité de compromis entre qualité des solutions et temps de calcul,
- ⊗ Existence de preuves de convergence asymptotique pour certaines méthodes,
- ⊗ Optimum non garanti,
- ⊗ Nécessité d'adaptation de la méthode au problème traité,
- ⊗ Nécessité de réglage des paramètres,
- ⊗ Difficulté de prévoir la performance.

4.4. Conclusion

L'optimisation est un sujet capital dans plusieurs domaines de recherche. Un grand nombre de problème peuvent en effet être décrits sous la forme de problème d'optimisation. Les problèmes d'ordonnancement, l'apprentissage supervisé de réseaux de neurones, ou encore la recherche du plus court chemin sont par exemple des problèmes d'optimisation.

Les méthodes exactes (déterministes) et les méthodes approchées (non déterministe) sont deux grandes approches utilisées pour résoudre ce type de problème. Les méthodes déterministes sont utilisées avec des problèmes de petites tailles tandis que les méthodes non déterministes résolvent les problèmes NP complet pour lesquels aucune méthode exacte n'est, à ce jour, découverte.

Nous avons présenté différentes méthodes d'optimisation et nous nous sommes, ensuite, focalisés sur l'optimisation avec essaim particulaires. Le principe de PSO est

Chapitre 3 : Méthodes d'optimisation

expliqué dans ce chapitre ainsi qu'un survol des travaux existants. Nous avons présenté un tableau récapitulatif qui englobe les différents problèmes d'optimisation où le PSO est utilisé.

4

CHAPITRE

Tâches d'exploration et de couverture

4.1.Introduction

Les réseaux de robots et de capteurs sans fil (WSRN : Wireless sensor and robot networks) sont composés d'un grand nombre de nœuds qui doivent coopérer pour accomplir une ou plusieurs tâches simultanément ou non (détecter la présence d'un intrus malveillant dans une région d'intérêt, surveiller l'occurrence d'un événement spécifique, etc.). Ils ont été appliqués dans des applications militaires ainsi que dans la surveillance de l'environnement et le domaine de l'agriculture intelligente. Un scénario typique d'application pour ces réseaux (WSRN) peut comporter deux phases : d'abord, le groupe occupe un environnement aussi rapidement que possible. Une fois que les nœuds (robots et/ou capteurs) sont distribués dans l'environnement, ils accomplissent des tâches telles que l'exploration, la cartographie d'un environnement ou la détection des intrus. La couverture de l'environnement est une question fondamentale dans l'accomplissement de telles tâches. La couverture est importante pour déterminer à quel point l'environnement cible est surveillé et un service est fourni (assuré).

Ce chapitre a pour vocation de survoler le domaine de l'exploration et de la couverture afin de se constituer une idée claire sur les problèmes qui y résident, tout en faisant le tour des solutions qui relatent les questions les plus prometteuses vers la maîtrise de ces deux problèmes, organisées en classes.

4.2.Problème d'exploration

4.2.1. Introduction

L'exploration de l'environnement est probablement l'un des problèmes les plus étudiés dans le champ de la robotique en essaim. Sempé [113] a divisé la tâche d'exploration en trois parties : la dispersion, la couverture et la localisation de cibles (voir figure 4.1).

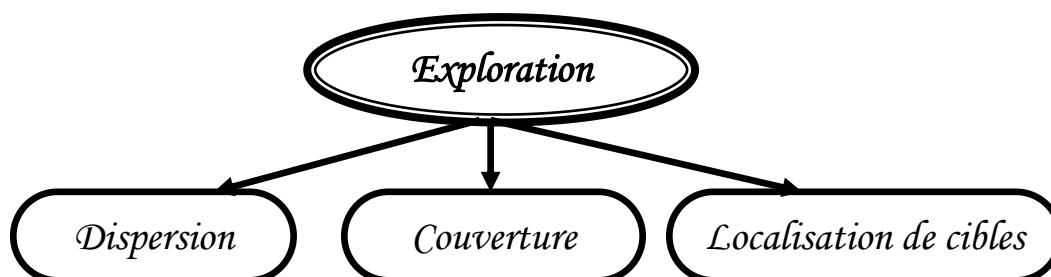


Figure 4.1 les trois parties de la tâche d'exploration

La dispersion correspond à une répartition spatiale des robots afin de couvrir au mieux une aire donnée. La façon dont l'aire est couverte par les robots peut varier avec le profil du terrain et la tâche à accomplir comme il est présenté dans l'expérience de Schwager et al. [114] où la répartition de robots dans l'environnement doit suivre la répartition d'une caractéristique de l'environnement. L'objectif est que la densité des robots soit plus importante dans les aires où cette caractéristique

Chapitre 4 : Tâches d'exploration et de couverture

environnementale s'exprime majoritairement. L'algorithme distribué proposé par *McLurkin* et al.[115] est un exemple à considérer dans cette partie. Grâce à cet algorithme, on peut obtenir une dispersion uniforme des robots dans des environnements complexes.

La couverture spatiale correspond à l'exploration systématique de l'ensemble des points d'une région. Les algorithmes de couverture trouvent leurs utilités dans des tâches aussi diverses que le déminage, l'inspection et le nettoyage ou encore la tonte du gazon.

La localisation de cibles permet de retrouver un ou plusieurs objets dispersés dans l'environnement. Cette tâche peut avoir de nombreuses variantes, dépendantes de la nature de la cible et de la manière de la localiser. Une étude expérimentale a été menée par [116] où il s'intéresse à la localisation d'une source d'odeur par un essaim de robots ; un robot qui a détecté une odeur émet également un signal qui attire vers lui ceux qui sont encore à la recherche d'une trace odorante. Grâce à ce recrutement, la probabilité de retrouver la source de l'odeur augmente.

4.2.2. Localisation et cartographie

La localisation et l'exploration sont des problèmes ouverts. Concernant la localisation, chaque robot doit être capable de connaître sa position actuelle ainsi que celles de ses voisins immédiats [117]. La localisation peut être relative (si le repère de référence est celui du robot dans sa situation initiale) ou absolue (si le repère de référence est lié à l'environnement)[118]. L'exploration est un problème commun à plusieurs applications robotiques. Lorsqu'un robot est employé pour explorer une zone inaccessible à l'homme, il doit être capable de fournir des informations à l'utilisateur sur la topologie des lieux visités et mettre à jour ses connaissances sur l'environnement. Il faut donc qu'il construise une carte de son environnement (mapping). Il doit en outre faire preuve d'une autonomie suffisante pour pouvoir explorer la zone et trouver son chemin même si les communications sont interrompues. Le robot doit donc être capable de se localiser en permanence dans l'environnement qu'il explore [119].

Les problèmes à traiter lors de la localisation sont l'autonomie d'un robot et la localisation coopérative. Pour cela la localisation a besoin de la cartographie. Le problème de l'autonomie d'un robot dans la tâche d'exploration d'un lieu inconnu s'appelle SLAM (**S**imultaneous **L**ocalisation and **M**apping). SLAM permet de corriger les erreurs accumulées pendant la phase d'estimation de mouvement en vue d'obtenir un modèle cohérent de l'environnement. Il traite ces problèmes de trois manières principales à savoir : l'**EKF – SLAM (Extended Kalman Filter SLAM)**, les **filtres particuliers** et les **graphes** [119]. SLAM a fait l'objet de plusieurs recherches. Roumeliotis et Bekey.[120] ont effectué une localisation multi-robots distribués en décomposant le filtre de Kalman étendu (EKF) en un certain nombre de filtres qui peuvent effectuer l'étape de prédiction de l'EKF localement sur chaque robot. Howard [121] a étudié l'exécution de SLAM, dans laquelle chaque robot est inconscient aux attitudes initiales des autres robots et commence un état d'estimation

Chapitre 4 : Tâches d'exploration et de couverture

de l'autre de manière décentralisée. Quand les robots se rencontrent pour la première fois, leurs cartes individuelles sont combinées en une carte commune utilisant des poses relatives. Le processus de cartographie est ensuite continué via la diffusion des nouvelles observations entre les robots. *Drouilly*[119] montre qu'il est possible de construire en temps réel des cartes en 3D très précises des environnements. Par ailleurs il peut extraire en temps réel des données à partir de la carte fournie par le SLAM, dans le but de fournir au robot des informations sur les objets présents dans la scène.

Concernant la localisation coopérative, ce problème est étudié par Kurazume et Hirose [122] qui ont introduit une méthode de localisation coopérative, où un réseau de robot est divisé en deux groupes. Un groupe de robot mobile, tandis que l'autre est stationnaire ; servant essentiellement de points de repère pour le premier groupe de robots pour la localisation. *Dieudonne* [123] traitent la localisation d'un réseau de robot comme un problème NP-difficile. Deux approches majeures ont été développées pour résoudre le problème de la localisation à savoir **la trilatération** (chaque nœud connaissant les positions de trois de ses voisins peut se localiser par l'intersection de trois cercles) et **la multilatération** (le même principe que la trilatération, en utilisant plus que trois points de référence) [124]. Dans ces approches, l'algorithme de localisation est implémenté selon l'organisation de calcul soit centralisé soit décentralisé. Dans les algorithmes centralisés, les nœuds capteurs recueillent des informations de leur environnement et les transmettent à une station de base qui à son tour les analyse, calcule les positions et les transmet aux nœuds). Par contre, dans les algorithmes distribués, tous les nœuds communiquent avec leurs voisins pour estimer les distances et échanger les informations de voisinage, afin de dériver leur position. A la fin du processus de localisation, chaque nœud doit connaître sa position ainsi que celles de ses voisins sans l'aide d'aucune unité centrale. Leung et al. [125] ont présenté un algorithme d'estimation décentralisé qui garantit le continuum de la mission dans un réseau de robots dynamique sans aucune condition sur la connectivité.

Malgré le nombre important de recherches ayant traité le problème de localisation, le critère de précision rend ce problème toujours ouvert.



Figure 4.2. Grille d'occupation d'un couloir de laboratoire



Figure 4.3. Représentation topologique

Comme on a mentionné ci-dessus, la localisation a besoin de la cartographie. La localisation a besoin, sous une forme ou une autre, d'une représentation informatique de l'environnement. La carte soit sous forme de grilles d'occupation (figure 4.2), soit de cartes topologiques (figure 4.3)[113]. Dans les grilles d'occupation, l'environnement est quadrillé en cases d'une dizaine de centimètres de côté. Durant l'exploration, chacune de ces zones se voit attribuer une probabilité d'être occupée par un obstacle. La localisation est sous forme de coordonnées qui désignent une case du quadrillage. L'exploitation de ces représentations est coûteuse en temps de calcul à cause de la grande quantité de données que les grilles doivent manipuler. Les cartes topologiques ne s'intéressent qu'aux points clefs du milieu. Elles ont la forme d'un graphe où les sommets représentent des endroits spécifiques (intersections de couloirs, changement de classe de perception...) et les arcs l'existence d'un chemin entre deux sommets.

4.2.3. Problèmes liés à la tâche d'exploration

4.2.3.1. Maintien de la connectivité

La tâche d'exploration exige un déploiement de robots et une communication fiable entre les membres de réseau. Par conséquent, la topologie du réseau peut varier de façon rapide et surtout imprévisible. Il s'agit de trouver une solution qui satisfasse au mieux deux contraintes contradictoires. D'une part, les robots doivent s'éloigner les uns des autres autant que possible afin d'accélérer l'exploration. D'autre part, ils doivent rester en contact les uns avec les autres pour échanger les données collectées (fragments de cartes) et coordonner leurs déplacements [126]. Le problème de maintien de la connectivité ainsi que d'autres problèmes font le sujet de beaucoup de recherches ; un état de l'art sur les travaux concernant les réseaux de robots qui tentent de résoudre ces problèmes sera présenté dans ce qui suit.

La mobilité et les pannes fréquentes causent le changement de la topologie, cette dernière exige le maintien de la connectivité. Le problème du maintien de la connectivité consiste à assurer l'existence d'un canal de communication fiable tout au long de la mission [126], les solutions proposées pour résoudre ce problème sont soit basées sur le contrôle de la topologie et la mobilité, soit celles qui rendent le robot conscient de la connectivité du réseau.

Les travaux ayant pris en considération les capacités limitées des équipements mobiles en termes de batterie pour le contrôle de topologie, utilisent la consommation d'énergie comme métrique et sont basés sur l'ajustement de la puissance de transmission des nœuds [127]. Ramanathan[128] et Lloyd [129] présentent des algorithmes de contrôle de topologie centralisés où une entité centrale calcule la puissance de transmission en utilisant la position des nœuds du réseau afin de réaliser une topologie avec une forte connectivité. Concernant les algorithmes distribués, Wattenhofer[130] montrent que les nœuds mobiles ajustent leur puissance de transmission suivant des informations locales pour maintenir un nombre défini de voisins. Meraihi[127] proposent de contrôler la morphologie du réseau en utilisant la mobilité d'un ensemble de routeurs dédiés, dans

Chapitre 4 : Tâches d'exploration et de couverture

le but d'offrir une meilleure connectivité. La nouveauté est d'utiliser positivement la mobilité, contrairement aux approches de contrôle de topologie classiques dans lesquelles la mobilité est subie.

Le contrôle de la mobilité a été traité par Basu [58] en proposant des algorithmes simples de contrôle de mouvement d'un sous ensemble de robots mobiles pour maintenir une "bi-connectivité", Cette étude se focalise sur des aspects de graphes afin de remédier au problème de partitionnement plus qu'à des aspects de performances réseau. Zavlanos[59]présentent l'idée de conjointre le contrôle de la mobilité et la communication pour faciliter la conception efficace des systèmes autonomes mobiles. D'après Tuan Le [60] la solution au problème de maintien de la connectivité dans un réseau de robot consiste en deux étapes principales ; la première est de fabriquer des robots conscients de la connectivité réseau et la deuxième c'est l'utilisation de ces connaissances pour planifier les tâches des robots sans compromettre la connectivité. Chaque robot dans le système doit bien réfléchir avant de se déplacer pour éviter de séparer la topologie du système de robots. Pour construire la topologie du réseau, chacun ne transmet que deux messages. Avec cela, le nombre de messages transmis diminue notablement, mais le réseau est bien construit. Le surcoût de communication est donc très bas. De plus, cela diminue la consommation d'énergie du robot et le temps nécessaire pour l'initialisation du réseau des robots. Une autre étude, de Tuan Le [126] concernant ce problème, est présentée dans sa thèse où il a proposé des contributions au maintien de la connectivité et à l'allocation dynamique de rôles. Ces travaux appliquent les techniques de MANET à la communication d'une équipe de robots mobiles. Parmi les travaux consacrés à l'étude des protocoles de routage dans un réseau MANET, on cite le travail d'Onat Yazır[131] qui proposent un routage adaptatif dans les réseaux mobiles ad hoc. L'idée consiste à commuter d'un protocole de routage vers un autre lorsque certaines conditions changent.

4.2.3.2.Minimisation du temps d'exploration

La minimisation du temps écoulé est un autre critère considéré lors de l'exploration d'environnement. Il existe des approches d'exploration qui se concentrent sur la réduction du temps nécessaire pour réaliser de telle tâche. Certains algorithmes garantissent une couverture complète, tandis que d'autres (notamment les algorithmes non déterministes) n'offrent pas une telle garantie, mais ils peuvent entraîner une charge de calcul diminuée par rapport aux algorithmes déterministes.

Les approches qui cherchent à minimiser le temps d'exploration peuvent être distinguées en deux classes : les approches qui ont le but de couvrir rapidement la plus grande surface et celles qui cherchent l'exhaustivité de l'exploration le plus tôt possible. Dans les premières la contrainte du temps doit être respecté même que l'environnement n'est pas exploré entièrement. Les secondes peuvent, par exemple, chercher à explorer des zones pour éviter d'y revenir ultérieurement. Un problème proche de la couverture exhaustive est la patrouille qui consiste à répéter des couvertures en minimisant le délai entre deux visites de la même zone [132].

4.2.3.3. Minimisation d'énergie

Généralement les robots mobiles sont dotés d'une quantité d'énergie limitée. Pour cela la consommation d'énergie doit être limitée pour une tâche d'exploration avec un environnement de grande taille [133]. Malgré cela, l'utilisation des moteurs est en général la principale source de consommation d'énergie ; auquel cas l'énergie consommée est proportionnelle à la distance parcourue et proportionnelle au temps d'exploration [132].

4.2.4. Travaux sur le problème d'exploration

Howard et al.[134] ont décrit la conception et la validation expérimentale d'un groupe de robots mobiles hétérogènes construit de DARPA SDR (SDR : Software for Distributed Robotics) PROGRAM. L'objectif principal du programme SDR est la conception et la mise en œuvre d'un système multi-robots autonomes qui pourraient explorer et cartographier un grand environnement intérieur, de déployer un réseau de capteurs dans cet environnement et d'utiliser ce réseau pour suivre les intrus et envoyer les informations récoltées à un opérateur distant. Ils ont construit une équipe hétérogène d'environ 80 robots. Avec une équipe de cette taille, le coût est un facteur important, les robots sont donc divisés en deux classes distinctes, avec des chiffres très différents, des capacités et des coûts: Un petit nombre de robots très performants (et coûteux), chacun équipé d'un laser à balayage télémètre, appareil photo, un CPU puissant et un grand nombre de robots relativement simples (peu coûteux), chacun équipé d'un microphone, caméra brut et un minimum de CPU à bord.

Pour la phase de cartographie, l'algorithme utilise deux composants centralisés et décentralisés: chaque robot utilise un algorithme incrémental de cartographie et de localisation simultanée à bord (SLAM) pour maintenir une estimation locale de position; ces estimations sont transmises à l'opérateur distant, où elles sont combinées à travers un second algorithme de SLAM pour générer les estimations cohérentes globales de position pour tous les robots. Par conséquent, une carte d'environnement, qui va être utilisée pour des opérations ultérieures de déploiement, est construite par la combinaison des données de tous les robots performants.

Pour l'exploration d'environnement, ils utilisent une approche décentralisée basée sur les frontières avec des grilles d'occupation locales et de la communication minimale entre robots. L'algorithme de base est le suivant :

1. Construire un plan de grille d'occupation locale en utilisant les données de télémétrie laser et les estimations locales.
2. Extraire une liste de frontières distinctes, à savoir, les frontières entre les régions connues et inconnues de la grille d'occupation. Jeter les frontières qui sont inaccessibles.

Chapitre 4 : Tâches d'exploration et de couverture

Il s'agit d'un processus à plusieurs étapes dans lequel les cellules de la frontière individuelle sont d'abord identifiées puis regroupées en composantes connexes. Les frontières inaccessibles sont détectées par la construction de l'espace de configuration de la grille d'occupation et en appliquant l'algorithme de Dijkstra. Si la frontière sélectionnée a disparu ou elle est devenue inaccessible (due à une obstruction par un autre robot), choisir au hasard une nouvelle frontière. Howard et al [134] critiquent leur algorithme et ils voient qu'il est simple mais en même temps il a plusieurs points forts : il est entièrement décentralisé (robuste à des défaillances de communication), ne nécessite pas une estimation globale de positions (robuste aux pannes ou des retards d'opérateur distant) et il est capable de résoudre les conflits des actions de robots sans communication explicite. D'autre part, cet algorithme souffre de deux problèmes qui sont la répétition d'exploration des zones à cause de la vue locale d'environnement et le manque de coordination explicite entre les robots qui peut mener à une exploration redondante.

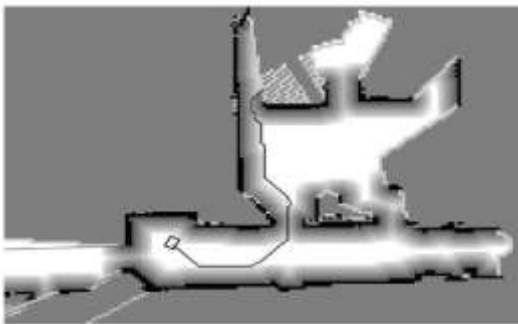


Figure 4.4 Aperçu de la carte locale utilisée pour l'exploration : Cellules noires représentent l'espace occupé, les cellules blanches sont libres et celles grises sombres sont des cellules inexplorées.

Witkowski et al, [135] présentent la méthode de triangulation dynamique qui est utilisée pour le déploiement des robots sur l'environnement. Les robots doivent être déployés dans le but de faciliter la communication et l'exploration de l'environnement. Pour pouvoir maîtriser la complexité géométrique de l'environnement, quelques robots peuvent être placés comme des ancres sur les entrées qui peuvent être des portes, début d'un passage et/ou placés sur les jonctions. Les autres robots sont distribués uniformément afin de collecter des informations sur l'environnement.

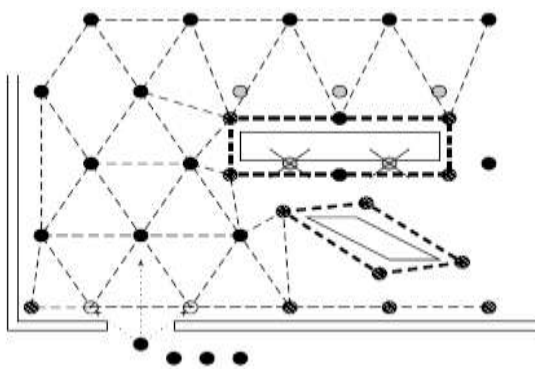


Figure 4.5 Méthode de triangulation dynamique : les points en blanc représentent les ancres placés dans l'entrée du site, les points en gris montrent les positions des robots dans le cas d'absence d'obstacles, les points en noir indiquent les robots loin aux ancres et les lignes en gras représentent les obstacles.

Chapitre 4 : Tâches d'exploration et de couverture

Les robots divisent l'environnement en régions (voir figure 4.5) formant une partition d'environnement, ces partitions vont représenter une triangulation dans le cas où l'environnement est sans obstacles. Cette partition doit être ajustable selon la mobilité des robots, l'ajout de nouveaux robots ou leurs pertes. Si un nouveau robot entre au site, il n'a pas besoin de chercher sa position, mais il peut remplacer le nœud le plus proche qui va à son tour remplacer son plus proche voisinetc. donc, le développement d'environnement se fait par des simples remplacements locaux des nœuds. Dans la figure 4.5, le nouveau robot peut se déplacer vers les trois possibilités indiquées par des flèches en points. Le protocole DSDV est modifié pour s'adapter au changement de topologie (l'ajout ou la perte d'un nœud).

4.3.Couverture

4.3.1. Zone de couverture

Un capteur possède deux zones : une zone de perception (rayon de capture R_s) et une zone de communication (rayon de communication R_c). La figure 4.6 illustre la zone de couverture et la zone de communication d'un capteur.

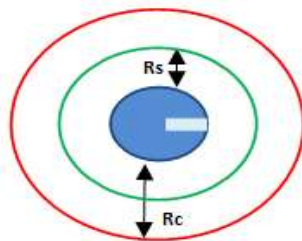


Figure 4.6 Portées d'un robot- R_s est la portée de détection tandis que R_c représente la portée de communication pour chaque robot.

4.3.2. Taxonomie de types de couverture

Les types de couverture sont différenciés selon plusieurs critères.

4.3.2.1.Surveillance de l'environnement

En se basant sur la façon dont un ensemble de capteurs assurent la surveillance d'une zone géographique. Erdelj et al. [136] distinguent trois types de couverture :

4.3.2.1.1. Couverture de zone

L'objectif principal est de surveiller (couvrir) une région donnée par un réseau de capteurs [48]. La figure 4.7 montre un exemple d'un déploiement aléatoire des capteurs pour couvrir une région.

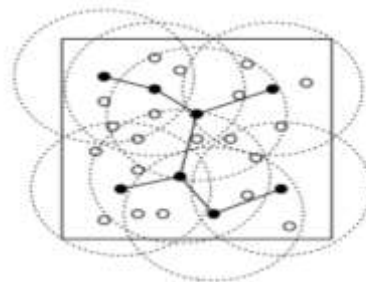


Figure 4.7 Couverture de Zone : des capteurs sont déployés aléatoirement sur une zone carrée pour la couvrir. Les cercles représentent les rayons de surveillance des capteurs, tout point de la zone est couvert par au moins un capteur.

4.3.2.1.2. Couverture de points

Dans le problème de couverture de point, l'objectif est de couvrir un ensemble de points spécifiques. La figure 4.8 montre un exemple d'un ensemble de capteurs déployés aléatoirement pour couvrir un ensemble de points (carrés en couleur bleu). Les nœuds noirs connectés forment l'ensemble des capteurs actifs.

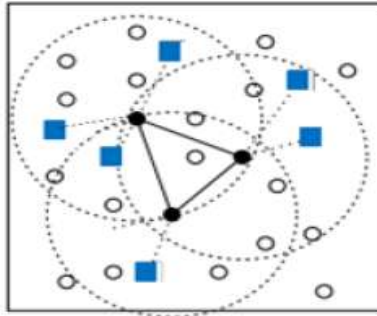


Figure 4.8 Couverture des points d'intérêt : on déploie des capteurs pour surveiller un ensemble de points spécifiques (cibles), dont la position géographique est connue.

4.3.2.1.3. Couverture de frontière

Afin de couvrir une partie de la zone d'intérêt, on a besoin seulement de couvrir le périmètre de cette région. C'est le problème de la couverture de frontière (barrière) (figure 4.9) [64].

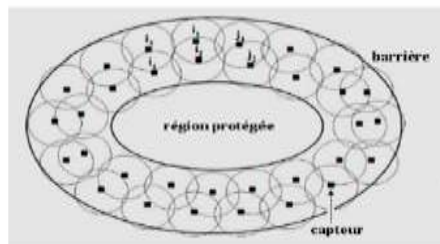


Figure 4.9 Couverture de frontière

4.3.2.2. Types de couverture

Selon la nécessité de précision de données et de fiabilité demandées par les applications en termes de précision des données et de fiabilité, deux types de couverture peuvent être distingués : couverture simple et couverture multiple.

4.3.2.2.1. Couverture simple

Dans une couverture simple chaque cible ou un point dans la zone doit être surveillé par au moins un capteur. Ce type de couverture est utilisé pour les applications qui ne nécessitent pas un grand degré de précision tel que la surveillance des champs agricoles [137].

4.3.2.2.2. Couverture multiple

En couverture multiple, chaque cible ou point dans la zone doit être surveillé par au moins k différents capteurs, qui s'appelle k -couverture de problème pour la zone de couverture.

Chapitre 4 : Tâches d'exploration et de couverture

4.3.2.3. Dynamisme de la couverture

La couverture peut être statique ou dynamique [138]:

4.3.2.3.1. Couverture statique

Dans ce type, chaque point dans l'environnement est couvert par un nœud à tout instant. Pour cela, il doit avoir un nombre suffisant (dépendant de la taille de l'environnement et de la portée de capture). Le problème c'est qu'avec un environnement inconnu, il est difficile, voire impossible, de déterminer le nombre de capteurs nécessaires.

4.3.2.3.2. Couverture dynamique

De l'autre côté, la couverture dynamique repose sur la mobilité des robots avec une configuration particulière, la couverture d'environnement peut être réalisée au fil du temps et avec n'importe quel nombre de robots.

4.4. Conclusion

Les problèmes de couverture, d'exploration et de cartographie sont des questions fondamentales dans le domaine de la robotique mobile. Pour les problèmes de couverture et d'exploration, l'objectif est d'utiliser un ensemble de robots mobiles équipés par des capteurs afin de couvrir de façon optimale tous l'environnement. Dans la cartographie, l'objectif est de construire une représentation probabiliste du monde basée sur des mesures obtenues par les capteurs. Notons au passage que la cartographie peut être utile aussi bien à la tâche d'exploration qu'à celle de couverture. L'exploration et la couverture sont deux concepts très liés. Nous avons consacré ce chapitre pour étudier ces deux problèmes. Dans la première partie, nous avons présenté le problème d'exploration et expliqué les concepts de localisation et de cartographie. Puis, nous avons étudié les différents critères (connectivité, temps et énergie) qui sont pris en considération lors de l'exploration d'environnement. Un survol de travaux de ce problème a été présenté. La deuxième partie concerne le problème de couverture où nous avons présenté une taxonomie de types de couverture.

5

CHAPITRE

Problème posé et solution proposée

5.1.Introduction

Un réseau de robots mobile est composé d'une collection distribuée de robots, ayant chacun des capacités de détection, de calcul, de communication et de mobilité employés dans le but d'accomplir des tâches coopératives tels que la localisation et le routage de messages. La mobilité et les pannes fréquentes causent le problème de déconnexion du réseau. La résolution de ce problème au sein d'un réseau de robots mobiles autonomes pour assurer la couverture où l'exploration d'environnement est primordiale. Le routage adaptatif se présente comme la réponse la plus appropriée, actuellement, à cette question.

5.2.Problème posé

Soit un système de N robots autonomes mobiles connectés via un medium radio, constituant un réseau ad hoc mobile sans fil (*MANET*). Ces robots sont envoyés en mission d'exploration sur un terrain dont les bordures sont connues pour découvrir M points d'intérêt(POI) dont les emplacements sont inconnus (voir Figure.6.1). Pour garantir un plus grand succès de la mission, les N robots coopèrent. Cette coopération dépend du système de communication construit sur un *MANET* que constitue ces robots. Généralement, deux critères d'arrêt sont possibles :

1. découvrir le maximum de points d'intérêt en un temps donné T , ou
2. découvrir un nombre de points d'intérêt qui dépasse un certain seuil (au cas le plus favorable se sera tous les points d'intérêt).

Il s'agit, dans ce cas, de trouver la meilleure façon de coopérer entre ces N robots pour explorer le terrain en question de la façon la plus efficace possible. La coopération est concrétisée par un échange de messages associés, éventuellement, à des actions ou comportements de base; deux aspects sont importants dans ces échanges :

1. ils doivent parvenir correctement et dans les délais à leur destinataire, et
2. ils doivent exprimer le niveau de coopération le plus haut possible.

Le premier aspect sera concrétisé par des communications fiables (le moins de déconnexions possibles avec une exploitation optimale de la bande passante). Quant au second, il doit être représenté par un jeu de règles manipulant des messages et des actions favorisant au plus haut degré possible le travail collectif conduisant à l'exploration, précité, la plus performante. Ceci revient à traiter un problème inverse où l'objectif à atteindre est préfixé ; il s'agit de l'optimisation de l'exploration collective d'un terrain donné en vue de couvrir M points d'intérêt. Il faudra, alors, trouver les facteurs ou les paramètres les plus significatifs, qui l'influencent (par émergence) le plus cette exploration collective et de rechercher leurs valeurs optimales dans l'espace des possibilités qu'ils constituent ensemble.

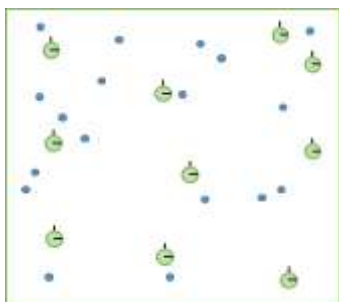


Figure 5.1 L'environnement d'exploration : M points d'intérêts sont dispersés de façon aléatoire sur un terrain dont les limitations sont connues. La bordure peut être de toutes formes fermées possibles. N robots sont aussi dispersés initialement de façon aléatoire.

5.3. Analyse du problème

5.3.1. Coopération

La coopération émerge au niveau macroscopique (niveau du groupe) à partir des actions des robots au niveau microscopique (niveau individuel) coordonnées par des messages qu'ils échangent via un réseau *MANET*. La qualité de cette coopération dépend de :

- La qualité du système de communication adopté dans sa globalité (la capacité du système de communication choisi de faire parvenir correctement tous les messages à leurs destinations).
- Le degré d'efficacité des messages destinés à la coopération en observant, au niveau du groupe, à quel point ils arrivent à permettre aux robots de s'entraider pour accomplir la tâche d'exploration.

En somme, la communication influence la coopération qui possède, à son tour, un effet sur le résultat recherché. On peut dire que la coopération, recherchée, est d'autant plus intéressante que le résultat obtenu est de meilleure qualité et qu'il s'est déroulé dans de meilleures conditions de communication (étant donné que le réglage des conditions de communication nous permet d'améliorer l'aspect « échange de messages » de la coopération).

5.3.2. Conditions de communication et paramètres à régler

Nous pouvons obtenir un bon résultat d'exploration avec beaucoup de perte de paquets (ce qui implique beaucoup de tentative de reconstruction de routes), dus à un grand nombre de déconnexions et à un protocole qui ne sait pas comment s'adapter à de telles situations de transmissions qui se présentent devant lui. Ceci aura une répercussion directe sur le temps d'accomplissement de la mission en question. Pour obliger la mission à se dérouler dans les meilleures conditions de communication possibles nous allons chercher à retrouver tous les points d'intérêt dans un délai que nous allons tenter de réduire au mieux. Ceci va permettre de rechercher le meilleur système de communication possible relatant les meilleures conditions de transmission réalisables.

Les meilleures conditions de communication sont représentées par :

Chapitre 5 : Problème posé et solution proposée

- le **paramètre d'adaptation** du processus de transmission (protocole) aux différentes situations qui se présentent devant lui, et
- le **paramètre de coopération** des différents robots en vue de réussir l'exploration.

Ces paramètres vont permettre de retrouver le maximum de points d'intérêt (*critère de complétude* du processus d'exploration) dans le délai le plus court possible (*critère d'accélération* du processus d'exploration). Nous allons remplacer le critère d'accélération par un critère plus direct ; il s'agit de réussir à faire passer avec succès le maximum de messages de communication (*critère de fiabilité* du système de communication utilisé).

Donc, seuls les paramètres d'adaptation et de coopération sont à régler pour prétendre à un accomplissement de la tâche d'exploration de la qualité la plus haute et en un temps le plus court possibles. Ceci est justifié par le fait que l'algorithme d'exploration est préfixé par hypothèse (pour des raisons de simplicité. Donc, on ne peut rien améliorer dedans). De plus, les mécanismes de base concernant les deux classes des protocoles de communication adoptées (*MANET* pour des taux de déconnexion normal et *DTN* pour des taux de déconnexion fréquente) sont aussi préfixées par hypothèse pour la même raison précitée. Donc, on ne peut rien améliorer ni dans les mécanismes de découverte, de maintien et de reconstruction des routes pour les *MANETs*, ni dans le Store and Forward pour les *DTNs*. Les réglages des deux paramètres qui restent sont effectués en vue d'obtenir de meilleurs résultats concernant les critères de complétude et de fiabilité.

- **Paramètre de coopération** : Pour ce paramètre, il s'agit de retrouver le jeu de règles qui permet d'exploiter la collectivité des robots de la meilleure façon possible ; un échange de messages de coopération adéquat effectué à temps et une activation de comportements de base adéquate, effectuée aussi à temps.
- **Paramètre d'adaptation de la transmission** : Pour ce paramètre, il s'agit de retrouver le jeu de règles qui permet un passage adéquat d'un protocole de communication vers l'autre. Comme cité ci-avant, le paramètre de coopération influence les critères de complétude et le paramètre d'adaptation influence le critère de fiabilité.

5.3.3. Fonction objective

Le critère de complétude peut être évalué via un taux de couverture des points d'intérêt (TCPI) alors que le critère de fiabilité peut être évalué via un taux de transmission de paquets avec succès (TTPS). La formulation de la fonction objective, exprimant ces deux critères à maximiser, peut être exprimée comme suit :

$$FO = \text{Max} (TCPI * TTPS) \quad \text{Eq6}$$

Avec $TCPI = \text{nombre de points couverts} / \text{nombre total de points d'intérêt}$, et

Chapitre 5 : Problème posé et solution proposée

$TTPS = \text{nombre de paquets transmis avec succès} / \text{nombre total de tentatives de transmission de paquets}.$

5.3.4. Exemples

Bon nombre d'exemples illustrent le fait que la communication (horizontale/verticale) a un effet direct sur la coopération qui influence à son tour la qualité de la tâche d'exploration ciblée. Pour illustrer ceci, donnons quelques exemples.

5.3.5. Communication horizontale

- Un robot R_d qui découvre un point d'intérêt POI_i à la position (x_i, y_i) diffuse un message pour informer les autres robots de cette découverte (cela suppose que les autres robots doivent être connectés). Ca va permettre aux robots informés de gagner du temps, en éliminant cette position de leur plan de recherche. Le message diffusé joue le rôle d'un repousseur utilisé.
- Un robot R_d qui détecte dans sa portée un nombre important de points d'intérêt déduit que son voisinage peut être riches en points d'intérêts et diffuse, en conséquence, un message pour attirer les autres robots vers cette zone. Ce message joue le rôle d'un attracteur utilisé avec une force calculée comme suit :

$$f_{ri} = (h/M)*F' \quad \text{Eq7}$$

Où f_{ri} est la force d'attraction du message i et F' est la force d'attraction, que produit les M points d'intérêt groupés et h le nombre des points d'intérêt que R_d détecté.

- Un robot R_d bloqué quelque part diffuse un message d'avertissement au reste du groupe pour les prévenir d'éviter la zone où il se trouve.

Remarque : Cette diffusion est d'autant plus efficace, dans ces trois exemples, que le nombre de robots informés est grand.

5.3.6. Communication verticale

Faire parvenir les informations concernant les points d'intérêts découverts (emplacements, états, ...) à une station de base SB ou à un centre de décision CD donné fait partie de la mission qui incombe à la collectivité des robots considérée; plus tôt tous les paquets concernant ces messages arrivent à destination correctement et plus tôt la mission aura prise de l'avance pour se terminer avec succès. Pour ce faire, des routes sont construites en utilisant des robots comme des nœuds intermédiaires entre les robots voulant transmettre les informations de leurs prélèvements et la SB destination finale de ces informations. L'objectif est de faire parvenir ces informations le plus tôt possible et de façon correcte en vue de dresser une carte de la région par exemple. Vu que ces robots sont des machines parallèles multifonctionnelles, ils peuvent explorer le terrain (se déplacer et lire leurs environnement locaux par exemple) tout en transmettant ou recevant des messages.

Chapitre 5 : Problème posé et solution proposée

Ceci peut poser le problème des coupures de ces chemins de transmission; il suffit que deux robots voisins, impliqués dans un chemin actif de communication, sortent de leurs portés de transmission pour que la liaison se casse. Si les coupures sont fréquentes, le nombre de tentatives de reconstruction des routes sera aussi grand et la mission sera considérablement affectée (un protocole comme *AODV* ne pourra pas résister à un tel phénomène, il deviendrait très vite inopérant) ; les robots passeront leur temps à reconstruire les chemins de communication, toutes les informations récoltées seront alors perdues et les points d'intérêts supposés explorés doivent rentrer de nouveau dans le plan d'exploration (selon l'algorithme adopté). Une solution possible dans un tel cas consiste en la mémorisation des informations recueillies et leur transmission quand cela est possible. Ceci nous fait transiter d'une politique classique d'acheminement de l'information appliquée dans les *MANET* comme celle de l'*AODV* vers une politique bio-inspirée de la propagation virale comme celle du store and forward appliquée dans les *DTN*. Mais cette dernière n'est pas aussi sans problème. Pour s'en convaincre, considérons le cas où le nombre de points d'intérêts est très grand et la capacité de mémorisation de chaque robot est faible (ce qui est souvent le cas réel). Tant que les prélèvements effectués sur le terrain ne sont pas transmis correctement, ils persisteront en mémoire, ce qui peut conduire à son débordement et une perte des informations avant leur transmission. La mission considérée sera, alors, remise en cause. Une solution possible à ceci est d'utiliser un protocole de communication adaptatif permettant de transmettre les paquets lorsque les conditions de transmissions classiques sont vérifiées et de les stocker en vue de les diffuser à la première occasion de connexion qui se présente lorsque ces conditions ne sont pas vérifiées.

Ce problème d'optimisation bi-objectif cherche à maximiser le nombre de découverte de points d'intérêts tout en maximisant le nombre de paquets délivrés correctement à leurs destinations. Si tous les paquets arrivent correctement à leur destination cela veut dire que la coopération s'est déroulée dans les meilleures conditions possibles. La réussite du deuxième objectif est aussi importante que celle du premier objectif au succès de toute la mission.

- a) L'adaptation du système de communication aux situations produites par les robots lors de leur application de l'algorithme d'exploration => une fiabilité plus grande des transmissions=> Coopération performante (renforcée par des transmissions fiables),
- b) (a) + Coopération optimale =>Exploration performante (complète et rapide).

5.4.Solution

5.4.1. Ressources logiques utilisées

- Un algorithme d'exploration,
- Deux protocoles issus respectivement des classes des protocoles de communication
 - *MANET* (exemple : *AODV*)

- DTN (exemple : inondation)

5.4.2. Ressources physiques (Modélisation du robot)

N robots connectés où chaque robot représente une machine qui peut exécuter certaines tâches en parallèle avec les autres machines, elle dispose de :

- Senseurs pour lire son environnement local,
- Actionneurs pour se déplacer,
- Système d'émission-réception pour envoyer des messages vers les autres robots ou la *SB* et recevoir des messages des autres robots ou un ordre de la *SB*,
- Un système de décision qui permet de transformer :

- **les entrées :**

1. lecture de son environnement local (issues des capteurs),
2. les messages d'entrées (issues du dispositif de réception)

- **en sorties :**

1. comportements de base,
2. messages de sorties.

On considère, dans ce travail, que les robots ne possèdent pas de système de raisonnement ; ils sont plus réactifs que délibératifs. C'est pour cette raison qu'ils utilisent des tables dites de consultation faisant correspondre à chaque vecteur d'entrées dans l'une de ces tables à un comportement de base et des messages de sortie. Les messages d'entrées seront traités à chaque pas de temps au niveau de chaque robot.

5.4.3. Paramètre de la coopération

Dans le contexte de la coopération, un robot peut percevoir dans son environnement local (selon sa portée de captage) :

- des points d'intérêts,
- rien

Il peut percevoir aussi des robots ou des obstacles, mais ce genre d'informations n'est pas intéressant pour la coopération. L'information sur l'existence de robots dans la portée de communication (nœuds voisins) sera exploitée dans les communications (au niveau des protocoles utilisés pour le routage) et celle des obstacles au moment des déplacements (au niveau de l'algorithme d'exploration adopté).

D'un point de vue communication, un robot possède plusieurs types de files d'attente organisées en pair : (file d'attente des entrées pour les messages reçus, file d'attente des sorties pour les messages à émettre).

Les types de ces pairs de files d'attente dépendent de types des messages enfilés dedans :

Chapitre 5 : Problème posé et solution proposée

- **File d'attente des Attracteurs (FAA)**
 - Pour la file de sortie (**FAAS**): diffuser l'information (coordonnées, force d'attraction) concernant les points d'intérêts perçus dans la portée R_c en multicast,
 - Pour la file d'entrée (**FAAE**) : contient toujours un seul attracteur celui dont la force est la plus grande : une fois le message arrive on compare sa force d'attraction avec celle existante dans la FAAE si elle est plus grande on le remplace.
- **File d'attente des Repousseurs (FAR)**
 - Pour la file de sortie (**FARS**) : informer tous les autres robots en broadcast des emplacements des points d'intérêts déjà couverts par ce robot ou des emplacements dangereux où le robot est bloqué par exemple; Ceci va leur permettre d'enlever de leur plan d'exploration les coordonnées de ces emplacements.
 - Pour la file d'entrée (**FARE**) : prendre en considération tous les messages qui arrivent pour mettre-à-jour le plan d'exploration du robot concerné.
- **Paquets à retransmettre** : le nœud concerné joue le rôle d'un nœud intermédiaire ou nœud source des informations à respectivement émettre ou retransmettre.
Exemples: transmettre les informations en relation avec les points d'intérêt que le robot concerné vient de couvrir vers la SB, paquets entrants à diffuser ou à acheminer vers sa destination. Pour ce type de message, c'est les protocoles de communication adopté qui s'en charge. Ils représentent une forme de coopération qui est déjà prise en charge par ces protocoles.

La structure de la table de consultation concernant les règles sensorimotrices exprimant la coopération des robots en vue d'améliorer les performances de la mission qui leurs incombent est comme suit :

<Perception locale coopérative><état du robot> → <action>

Perception locale

Le robot peut percevoir $M(M \geq 1)$ points d'intérêts non encore couverts (il ne possède pas leurs coordonnées en mémoire). Comme il peut ne percevoir aucun ($M = 0$).

- **Au moins un point d'intérêt** (condition de déclenchement de la couverture),
- **Aucun** (pas de points d'intérêts dans cette zone).

États du robot

- **Bloqué** (le robot est coincé quelque part et ne peut pas bouger, ceci peut être momentané ou sa batterie devient très faible au point qu'il n'arrive plus à bouger ou son dispositif de locomotion est en panne, etc.),
- **Attiré** (Le robot est en mission prioritaire, il doit arriver à la destination qu'il lui a été indiquée).
- **Libre** (c'est le seul état où le robot peut utiliser l'algorithme d'exploration adopté).

Chapitre 5 : Problème posé et solution proposée

- **Trouvé** (le robot détecte des points d'intérêts non couverts dans sa portée de détection).

Actions

- **Couvrir** : le robot prélève les coordonnées (processus de localisation), et les états (processus de captage) des points d'intérêts qui se trouvent dans sa portée R_c non encore marqués (leurs coordonnées ne sont pas encore mémorisées. Il transmet ces informations à la *SB* via le protocole adopté (*MANET* ou *DTN*), diffuse en broadcast (à mettre dans *FARS*) un message pour que les autres robots mettent à jour leur plan d'exploration (répulsion), mémorise les coordonnées de ces points d'intérêts qu'il vient de localiser (marquage). Après, il calcul la force d'attraction des points qu'il vient de couvrir, si cette force dépasse un certain seuil il transmet un message en multicast (à mettre dans la *FAAS*) pour un certain nombre de robots pour les attirer vers sa région (attraction). Il met, en fin du compte, son état à « libre » et reprend l'exploration de nouveau.
- **Aller à** : le robot verrouille son état sur « attiré », il se dirige vers l'attracteur (potentiel) qu'il trouve dans sa *FAAE*, il se déplace sans exploiter l'algorithme d'exploration jusqu'à ce qu'il arrive à sa destination. Une fois il arrive il met son état à libre et reprend l'exploration).
- **Explorer** : le robot exploite l'algorithme d'exploration adopté, s'il n'arrive pas à se déplacer il envoie un message de type repousseur en broadcast pour avertir ses coéquipiers, d'éviter cet emplacement, en utilisant le protocole adopté et se met en état bloqué.
- **Ne rien faire**: Le robot reste dans le même état et il ne rien fait.

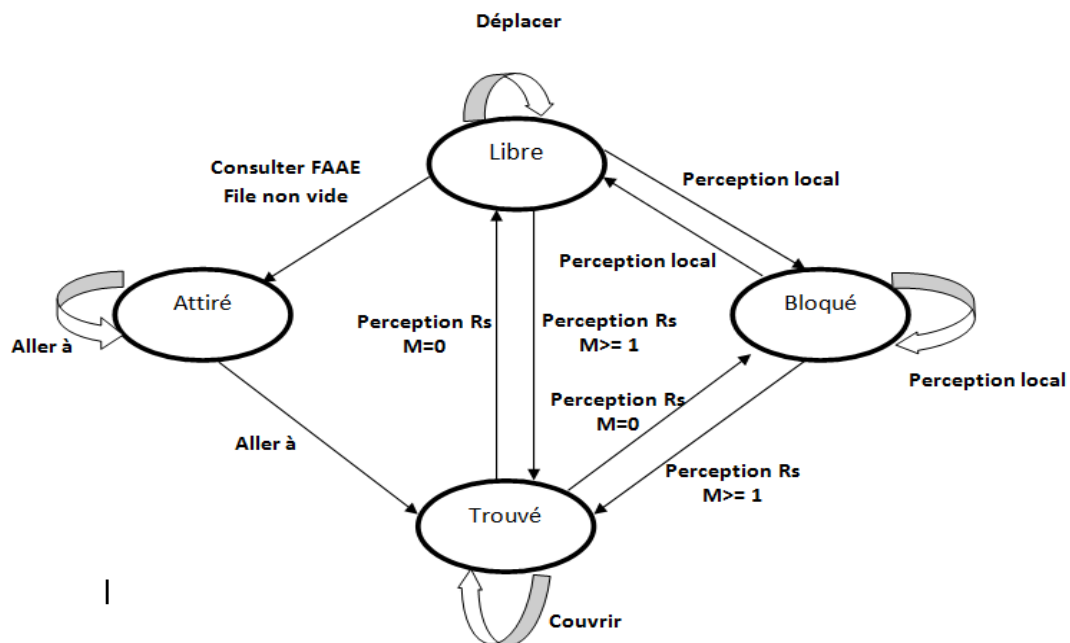


Figure 5.2 Diagramme de transition d'états

Chapitre 5 : Problème posé et solution proposée

- **Perception local** : le robot perçoit les trois cases qui l'entourent. Si toutes les cases sont occupées, le robot change son état à « bloqué » sinon il se déplace et il reste en état « libre ».
- **Perception Rs** : le robot peut percevoir des points d'intérêts non couverts ($M \geq 1$) dans sa portée de détection comme il peut ne percevoir aucun ($M = 0$) ou ils sont déjà marqués.

En parallèle, pour ces quatre actions, si la file *FARE* est non vide, le robot met à jour son plan d'exploration en prenant en compte tous les points d'intérêts déjà couverts ou les avertissements qui se trouvent dedans.

Aussi, dans tous ces cas en traite en parallèle (si possible) les paquets d'entrées et de sorties propres aux protocoles (MANET ou DTN).

Notons que dans cette approche, et pour des raisons de simplification, nous avons encapsulé un maximum de logique de coopération au sein des comportements de base, en ne laissant que l'essentiel à l'extérieur. Ceci représente un maximum de programmation en dure. Par la suite on aura l'occasion de ressortir de plus en plus de cette logique pour la paramétrer.

Taille de la table de consultation coopérative

La taille de cette table est : $4 \times 2 = 8$ entrées,

Le nombre des tables est : $4^8 = 65536$,

Le code des actions 00, 01, 10, 11.

En binaire les valeurs des sorties de ces tables varient entre 00 00 00 00 00 00 00 00 et 11 11 11 11 11 11 11 11 si on les converties en décimale ils représentent l'intervalle $[0, 2^{16}-1] = [0, 65536-1] = [0, 65535]$ qui représente l'axe du paramètre de coopération.

5.4.4. Paramètre de l'adaptation

Dans le contexte de l'adaptation, un robot peut transmettre selon deux modes différents :

- Le premier mode concerne la classe des protocoles *MANET* : ce mode est adéquat lorsque la densité des nœuds est relativement importante, ce qui évite les déconnexions fréquentes des routes établies (les nœuds actifs se trouvent fréquemment dans les portées de transmission des uns les autres et à la déconnexion une autre route est facilement trouvée).
- Le deuxième mode concerne la classe des protocoles *DTN* : ce mode est adéquat lorsque les robots se trouvent éparpillés. Le réseau des robots se verra, alors, partitionné en plusieurs sous-réseaux (cette situation peut ne pas être permanente).

La métrique choisie pour départager les deux modes est le taux local d'échecs (*TLE*). Cette métrique est locale dans le sens qu'elle concerne seulement les nœuds

Chapitre 5 : Problème posé et solution proposée

impliqués dans un chemin donné. Chacun de ces robots comptabilise le nombre de déconnexions par lesquels il est passé durant une période Δt . Si ce nombre est important il prend conscience que le réseau auquel il appartient vérifie les propriétés d'un DTN et il doit commuter en conséquence vers ce dernier, s'il se prend pour un MANET sinon il préserve son identité. Par contre, si ce nombre est au-dessus du seuil, il prend conscience que le réseau auquel il appartient vérifie les propriétés d'un MANET et doit commuter vers ce dernier s'il se prend pour un DTN sinon il préserve son identité.

Une prise de conscience collective et progressive peut émerger pour atteindre tout le réseau. Mais, il peut être que passagère. Donc, à tout instant on verra des sous-groupes de robots qui pratiquent des protocoles de communication appartenant à des classes différentes. Chaque groupe guidé par le contexte dans lequel se trouve son sous-réseau. Pour le protocole appartenant au MANET ce groupe doit être formée au minimum des nœuds actifs appartenant à un chemin donné. Pour un protocole DTN ils appartiennent à un sous-réseau isolé pour un certain temps.

Ce taux local d'échecs $TLE_i(\Delta t)$ concernant le robot i pendant le temps Δt est exprimé en fonction du nombre de tentatives d'établissement ou de rétablissement de routes non réussies où ce robot a été impliqué pendant Δt (NTNR) et le nombre total de tentatives par rapport à ce robot pendant Δt (NTT).

$$TLE_i(\Delta t) = NTNR/NTT \quad \text{Eq8}$$

En suivant le même raisonnement que pour la coopération, la table de consultation concernant l'adaptation à la structure suivante :

<Taux d'échecs de connexion><protocole courant> → <Action>

Taux d'échecs de la connexion :

- Au-dessus d'un seuil donné
- Au-dessous d'un même seuil

Protocole courant :

- DTN
- MANET

Action :

- Switcher
- Persister

La taille de la table est $2 \times 2 = 4$

Les codes sont 0, 1 et 0, 1

Le nombre de table est $2^4 = 16$

Chapitre 5 : Problème posé et solution proposée

Les valeurs des sortie dans chaque table vont de 0, 0, 0, 0 à 1, 1, 1, 1.

En décimale on retrouve l'intervalle suivant $[0, 2^4-1] = [0, 15]$ ceci représente l'axe du paramètre d'adaptation.

L'espace de recherche dans ce cas est $65535 \times 15 = 983025$ positions de particules.

En recherchant la solution via l'heuristique PSO, ça va nous donner une courbe qui ressemble à la suivante [84] :

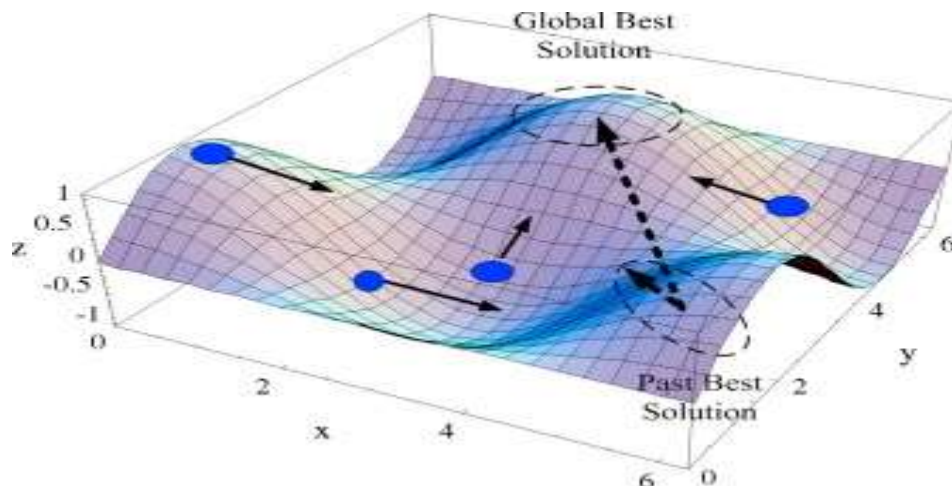


Figure 5.3 Solution via PSO : L'axe des x représente par exemple le paramètre de coopération et l'axe des y représente le paramètre d'adaptation et l'axe des z représente la fonction objective recherchée.

5.5. Particle Swarm Optimization (PSO)

PSO est une technique d'optimisation stochastique qui se base sur le principe de la population et appartient à la classe des approches heuristiques bio-inspirée. Plus précisément elle appartient à la catégorie des techniques de calculs informatique basées intelligence en essaim.

5.5.1. Principe

Notre problème est bidimensionnel. Ceci veut dire qu'on a pour chaque particule i (ou individu de la population à considérer) deux valeurs : - une selon le paramètre de la coopération X_1 , et l'autre selon le paramètre de l'adaptation X_2 (qu'on a appelé auparavant x et y). Initialement la particule i possède une position aléatoire (X_{i1}, X_{i2}) dans l'espace de recherche et une vitesse aléatoire (V_{i1}, V_{i2}) . Pour faire progresser cette méthode vers la solution optimale, deux valeurs sont sauvegardées par chaque particule i à chaque itération :

- Sa meilleure expérience propre est sauvegardée dans la position P_i et la valeur de la fitness P_{best} ,

Chapitre 5 : Problème posé et solution proposée

- La meilleure expérience de toute la population est sauvegardée dans la position P_g et la valeur de la fitness g_{best}

Toutes ces valeurs seront utilisées à chaque itération pour mettre à jour les positions et les vitesses de toutes les particules de la population considérée selon deux équations [85] :

$$V_i^{K+1} = w.V_i^K + C_1.rand_1.(pbest_i^k - x_i^k) + C_2.rand_2.(gbest_i^k - x_i^k) \quad \text{Eq4}$$

$$X_i^{K+1} = X_i^K + V_i^{K+1} \quad \text{Eq5}$$

Tel que : K est le nombre d'itération, w est le coefficient d'inertie, C_1 et C_2 sont les coefficients d'accélération, rand1 et rand2 sont des nombres aléatoires distribués dans l'intervalle [0,1].

5.5.2. Valeurs des paramètres du PSO utilisés

Les paramètres de l'algorithme de PSO sont choisis comme suit: nous utilisons une population de 50 particules avec 50 itérations.

5.5.3. Algorithme de PSO

L'algorithme de PSO est illustré par le pseudo code suivant:

Algorithme 1: algorithme de PSO

1. Initialiser l'essaim
2. Evaluer l'essaim
3. **Répéter** nombre d'itérations
4. **Pour chaque** particule i **faire**
5. Calculer fitness (xi) // **exécuter la simulation des robots**
6. **Si** fitness(xi) > fitness(pbest_i) **alors**
7. fitness(pbest_i) = fitness (xi)
8. pbest_i = x_i
9. **Fin si**
10. **Fin pour**
11. fitness (g) = max fitness (pbest_i)
12. **si** fitness(g) > fitness(gbest) **alors**
13. Fitness(gbest) = fitness(g)
14. **Pour chaque** particule i **faire**
15. gbest = pbest i
16. **Fin pour**
17. **Fin si**
18. Mise à jour vitesse de la particule
19. Mise à jour position de la particule
20. **Fin répéter**
21. **Retourner** gbest

5.6. Algorithme d'exploration

Cet algorithme est inspiré de celui proposé par Martijn et al. [139] basé sur deux étapes : la création d'une population et le calcul d'une valeur d'utilité qui facilite le déplacement des robots. Dans notre travail, l'algorithme proposé n'est pas appliqué sur tous les robots mais seulement avec un robot qui est en état libre. Les deux étapes de l'algorithme sont présentées dans ce qui suit.

5.6.1. Création de la population

A chaque instant t , chaque robot i possède une position $P_t^i = (X_t^i, Y_t^i)$ dans l'environnement.

- On appelle une configuration la position actuelle du robot i , on note $Cfg_t = \{P_t^i\}$. Un robot dans un état libre doit se déplacer aux différentes positions afin de trouver les points d'intérêt dispersés dans l'environnement. Le robot i perçoit son environnement local et exécute un mouvement m qui correspond à la perception effectuée. L'exécution d'un mouvement m par le robot i , lui permet d'arriver à une nouvelle position P_{t+1}^i .
- On note $Cfg-C_{t+1}$ la nouvelle configuration qui représente toutes les positions que le robot i peut occuper à l'instant $t+1$. On définit $Cfg-C_{t+1} = \{P_{t+1}^i\}$. On considère que le robot peut se déplacer vers l'une des trois (3) cases dans l'environnement locale qui l'entourent (voir Fig.6.5), dans le cas où le déplacement est possible sinon le robot garde sa position actuelle $P_{t+1}^i = P_t^i$ et aucun mouvement de déplacement ne sera effectué.

$$P_{t+1}^i \in P = \{W, N, E\}$$

Par conséquent, trois (3) configurations « $Cfg-C_{t+1}$ » sont distinguées pour un robot i

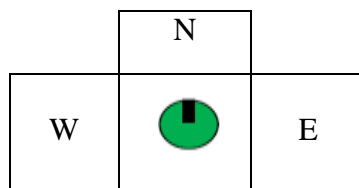


Figure 5.4 Les différentes directions considérées d'un robot dans notre Algorithme d'exploration

Une population qui regroupe les trois configurations possibles sera créée à chaque instant t .

$$\text{pop} = \{Cfg-c_{t+1}^1, Cfg-c_{t+1}^2, Cfg-c_{t+1}^3\} \text{ tel que } k = 3.$$

- Une troisième configuration notée $Cfg-M_{t+1}$ représente le mouvement exécuté par le robot i à l'instant $t+1$, $Cfg-M_{t+1} = \{m_{t+1}^i\}$. A l'instant $t+1$, le robot peut se déplacer vers l'une des trois cases ($m_{t+1}^i = D$) comme il peut garder sa même position et effectuer d'autres mouvements tels que le changement de son état ($m_{t+1}^i = C$) ainsi que l'envoi d'un message de type repousseur aux autres robots ($m_{t+1}^i = S$) si toutes les trois cases sont occupées.

Chapitre 5 : Problème posé et solution proposée

En conséquence, le mouvement m_{t+1}^i peut appartenir à l'ensemble M défini comme suit : $m_{t+1}^i \in M = \{D; C; S\}$

5.6.2. Calcul d'utilité

Après la création de la population, une valeur d'utilité est calculée pour chaque configuration et celle qui a la meilleure valeur doit être sélectionnée. Pour cela, lorsque le robot veut se déplacer, les situations suivantes sont considérées:

1. **Position occupée** : la position vers laquelle le robot veut se déplacer peut être occupée par un obstacle. On considère que l'obstacle peut être un autre robot ou un point d'intérêt. Dans ce cas le robot ne peut jamais se déplacer vers cette position.
2. **Position libre** : la position vers laquelle le robot veut se déplacer peut être libre, dans ce cas et afin d'éviter le problème de la boucle infinie, une fonction aléatoire choisit l'une des trois cases.

La fonction d'utilité UCE ($Cfg-C_{t+1}^i$) pour le robot i et qui prend en considération les situations expliquées ci-dessus, est définie comme suit :

$$U_{CE}(cfg-c_{t+1}^i) = \begin{cases} 0 & \text{Si position occupée} \\ 1 & \text{Si position libre} \end{cases}$$

Dans le cas où plus d'une case est libre, le robot choisit aléatoirement la direction vers laquelle il veut se déplacer.

Après le calcul d'utilité, la configuration qui a la valeur maximale est sélectionnée et le robot exécute le mouvement qui correspond.

Pour une configuration $cfg-c_{max}$ avec :

1. une utilité égale à 1 ; le robot se déplace vers la position P_{t+1}^i correspondante à $cfg-c_{max}$ et garde le même état « libre ». ($m_{t+1}^i = D$) et ($P_{t+1}^i = P_{t+1}^i$ avec utilité max).
2. une utilité égale à 0 ; le robot change son état à « bloqué » et envoie un message de type repousseur aux autres robots tout en gardant la même position. ($m_{t+1}^i = C + S$) et ($P_{t+1}^i = P_t^i$).

Algorithme2 : Algorithme d'exploration

1. $i=1$ $K=3$;
2. **Pour** $i= 1$ **to** k **do**
 // Générer une population des configurations « Cfg-C »
 // Chaque configuration correspond à une position
3. Random select $P^i_{t+1} \in P = \{N;E;W\}$
4. $cfg-c^i_{t+1} = \{P^i_{t+1}\}$
5. $i=i+1$
6. **Fin pour**
7. $Pop = \{cfg-c^1_{t+1}, cfg-c^2_{t+1}, cfg-c^3_{t+1}\}$
8. $cfg-c_{max} = cfg-c^1_{t+1}$
9. $i=1$ $K=3$;
10. **Pour** $i= 1$ **to** k **do**
11. Calculer $U_{CE}(cfg-c^i_{t+1})$
12. **Si** $U_{CE}(cfg-c^i_{t+1}) \geq U_{CE}(cfg-c_{max})$ **Alors**
13. $cfg-c_{max} = cfg-c^i_{t+1}$
14. **Fin si**
15. **Fin pour**
 // Exécuter m^i_{t+1} correspond
 // Changer la position P^i_{t+1}
16. **Si** $cfg-c_{max} = 0$ **alors**
17. $(P^i_{t+1} = P^i_t)$
18. $(m^i_{t+1} = C + S)$
19. **Fin si**
20. **Si** $cfg-c_{max} = 1$ **alors**
21. $(P^i_{t+1} = cfg-c_{max})$
22. $(m^i_{t+1} = D)$
23. **Fin si**

5.7. Protocoles de routage

5.7.1. MANET

Dans les réseaux MANET, le routage se base sur l'hypothèse que les nœuds peuvent communiquer seulement s'il existe un chemin debout-en-bout entre la source et la destination. Divers protocoles de routage ont été développés. Ces protocoles de routage peuvent être classés en des protocoles proactifs et ceux à la demande. Dans les protocoles de routage proactif comme DSDV et le protocole OLSR chaque nœud construit en permanence une table de routage qui contient tous les meilleurs chemins existants aux d'autres nœuds. Cette catégorie de protocoles a comme avantages le gain de temps lorsqu'une route est demandée, mais provoque l'occupation de la bande passante et la consommation importante d'énergie pour le maintien de la mise à jour de table de routage qui peut être énorme suivant les nœuds présents dans le réseau. Les protocoles de routage à la demande tels qu'AODV et DSR ont été proposées pour résoudre le problème de routage proactif ; la route est établie uniquement à la demande. Dans l'AODV, le nœud destinataire répond à la RREQ arrivée vers le nœud source et AODV conserve une seule entrée de routage pour chaque direction d'une route active.

Puisque notre objectif est de fournir des paquets dans le réseau de robots avec une topologie dynamique, les protocoles de routage proactifs ne sont pas souhaitables en raison de leurs inconvénients. Par conséquent, nous avons choisi AODV comme un

Chapitre 5 : Problème posé et solution proposée

protocole de routage réactif pour être utilisé par une équipe de robots connectés.

Le protocole AODV se base sur deux mécanismes «Découverte de route» et «Maintenance de route», il utilise les principes des numéros de séquence afin de maintenir la consistance des informations de routage. Trois types de message se déroulent dans le réseau Ad hoc : RREQ, RREP et RERR. Lors de découverte de route, le nœud source émet le message RREQ(ROUTE REQUEST) et déclenche un compteur en attendant une réponse. Les nœuds intermédiaires diffusent ce message jusqu'à atteindre sa destination, celui-ci génère un message RREP (ROUTE REPLY) destiné à la source. Quand un lien entre deux nœuds devient défaillant, le nœud source diffuse un paquet RERR (Route ERROR) afin de maintenir la route.

5.7.2. DTN

Le principal problème dans un MANET est la perte de paquets en l'absence de chemins debout en bout vers leurs destinations. Son objectif principal est la minimisation de délai ce que signifie la livraison des paquets le plutôt possible. Contrairement, les paquets dans le réseau opportuniste ou DTN sont stockés et inondés dans le réseau jusqu'à ce qu'ils atteignent leurs destinations. En outre, la maximisation de la probabilité de livraison est l'objectif principal de DTN. La majorité des stratégies de routage sont classés en fonction du nombre de copies du message envoyés, créés en des stratégies de transmission (forwarding), réplication (replication) et l'inondation (flooding). La stratégie de transmission consiste en la propagation d'un message à copie unique. Seule la source et la destination de ce message ont une copie. Avec une stratégie de réplication, un grand nombre de nœuds intermédiaires ont une copie du même message. Dans les stratégies de réplication extrêmes appelées stratégies d'inondation, chaque nœud du réseau a une copie du message envoyé. L'épidémique et MaxProp sont deux protocoles typiques de la stratégie d'inondation. C'est la mobilité des nœuds qui permet aux messages de se retrouver à-la-fin du compte chez leurs destinataires légitimes.

Dans notre travail, nous considérons que les messages sont envoyés en mode broadcast, en d'autres termes ; tous les robots sont des destinations de message. Pour des raisons de simplification, nous supposons que les robots communiquent par un routage épidémique dans la situation de déconnexion et aucune technique de gestion de la mémoire n'est appliquée à la différence de protocole MaxProp. Un pseudo code expliquant le principe de fonctionnement du routage épidémique est présenté dans ce qui suit.

Algorithme 3 : routage épidémique

Pour chaque message « m » faire

1. Le robot S diffuse le message vers ces voisins actifs (dedans sa portée de communication R_c).
2. Chaque robot reçoit le message :
 - a. Vérifie si le message existe déjà dans son tampon alors il l'ignore
 - b. Sinon il l'insère dans son tampon et il mémorise l'ID de message ainsi que sa source.
 - c. Il retire son ID de la liste des destinataires
 - d. Il répond par un ACK au robot source de ce message
 - e. Il diffuse le message vers ces voisins.
 - f. Echange de message avec les robots rencontrés

5.8.Algorithme de simulation

Après l'initialisation et l'évaluation de l'essaim, la simulation est lancée et les robots se comportent suivant la particule qui l'exécute. L'algorithme utilisé est représenté dans ce qui suit avec un nombre d'expériences égales à 10 et 500 pas de simulation:

Algorithme 4 : Algorithme de simulation de robots

1. Initialisation des paramètres des particules
2. **Répéter nombre-itération**
3. **Pour chaque particule**
4. Initialisation d'environnement
5. **Répéter nombre-expérience**
6. Changer l'environnement
7. **Répéter pas-simulation**
8. **Répéter DT**
9. **Pour chaque robot (1-nombre de robots)**
10. Percevoir l'environnement local (R_s)
11. Lire le comportement de coopération
12. Exécuter l'action correspondante à la perception
13. Si action= 00 alors couvrir
14. Si action= 01 alors explorer
15. Si action= 10 alors aller à
16. Si action= 11 alors exécuter ordre
17. Vérifier les files de sortie
18. Diffuser les messages suivant le protocole courant
19. Établissement des routes
20. Si route non établie alors
21. Marquer tentative non réussie (**TNR**)
22. **Fin pour**
23. **Fin répéter**

- 24. Pour chaque robot (1-nombre de robots)**
- 25.** Calculer TLE (**TNRs/ TT**)
- 26.** Lire le comportement d'adaptation
- 27.** Exécuter l'action correspondante aux paramètres d'adaptation (TLE + protocole courant)
- 28.** Si action= 0 alors persister
- 29.** Si action= 1 alors switcher
- 30. Fin pour**
- 31. Fin répéter**
- 32.** Comptabiliser le nombre de messages envoyés
- 33.** Comptabiliser le nombre de messages reçus
- 34.** Comptabiliser le nombre de POI couverts
- 35.** Évaluer la fitness (**E-fitness**)
- 36.** Accumuler les valeurs des évaluations dans $Fit = Fit + E\text{-fitness}$
- 37. Fin répéter**
- 38.** Évaluer la fitness moyenne dans $Fitness = Fit / \text{nombre-expérience}$
- 39.** Sauvegarder la fitness de la particule
- 40. Fin pour**
- 41.** Évaluer l'essaim
- 42.** Trouver la fitness maximal (**max Fitness**)
- 43. Fin répéter**
- 44. Fin**

5.9.Conclusion

On a consacré ce chapitre à l'étude du problème traité dans ce mémoire ainsi que la solution proposée. Afin de résoudre le problème de perte de message à cause de déconnexion dans un réseau de robots, nous proposons d'acheminer les messages d'une façon adaptative tout en assurant l'accomplissement de la tâche d'exploration.

Les différents algorithmes développés, les protocoles de routage choisis ainsi que la méthode d'optimisation ont été présentés.

6

CHAPITRE

Mise-en-œuvre et résultats de simulation

6.1.Introduction

La simulation est la reproduction d'un phénomène dans le but est de vérifier des théories et des hypothèses qui permettent d'expliquer un phénomène donné ensuite de prédire l'évolution de ce phénomène. Le processus de simulation peut être résumé en quatre étapes:

1. L'analyse et l'observation du système réel afin qu'on arrive à extraire les informations qui vont permettre de créer un modèle.
2. La représentation du modèle sous forme d'un programme écrit en langage machine.
3. L'exécution du programme développé dans le simulateur.
4. L'analyse des résultats obtenus nous permet de déduire des informations sur le modèle.

La simulation multi-agent est défini comme l'intersection de la simulation informatique et les systèmes multi Agents (SMA). Son objectif est de créer un mode virtuel composé d'un ensemble d'agents en interaction. Ces agents agissent et modifient l'environnement selon leurs règles de comportements. Dans notre cas d'étude il s'agit d'une simulation à base d'agents dont les agents sont les robots mobiles autonomes dispersés dans l'environnement qui coopèrent et communiquent dans le but d'explorer un ensemble de points d'intérêts.

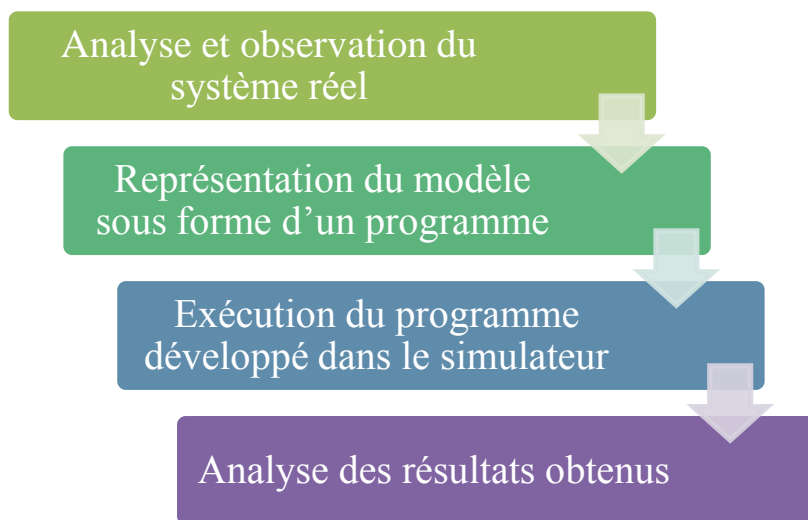


Figure 6.1. Processus de simulation

6.2.Mise en œuvre de la simulation

6.2.1. Choix de simulateur

Afin de concrétiser nos propositions, nous avons opté pour l'environnement de simulation orienté agents « Netlogo » version 5.0.5[140]. Ce logiciel est conçu en java ce qui le rend portable sur n'importe quelle machine. Il est adéquat pour simuler les phénomènes sociaux et naturels. En outre il est relativement simple à manipuler et

Chapitre 6 : Mise-en-œuvre et résultats de simulation

possède des outils intégrés (comme les illustrations graphiques) permettant un traitement facile des résultats. Le principal avantage de ce langage c'est qu'il permet la modélisation d'un système complexe composé d'un nombre infini d'entités autonomes opérants en parallèle et dispersés dans un environnement statique ou dynamique.

6.2.2. Mise en œuvre de la solution proposée

Le système présenté est un système distribué composé d'un ensemble de robots mobiles, homogènes et autonomes. On suppose que chaque robot:

- Est équipé par des capteurs qui détectent les événements produits dans sa portée de détection. La plage de détection est indiquée par un cercle de rayon R_s centré sur le robot.
- A une capacité limitée de communication avec des robots qui se trouvent dans sa portée de communication.
- Est équipé d'appareil de communication sans fil.
- A la capacité de locomotion et de distinguer le robot de la frontière.
- A une capacité de la localisation.
- Peut exécuter des tâches parallèles.

On dispose donc d'un ensemble des agents-robots et des objets répartis sur un environnement dont les bordures sont connues. Les robots ont une tâche commune, qui consiste à explorer un ensemble de points d'intérêts dont leur nombre est connu au préalable. Afin d'accomplir leur mission avec succès, les robots doivent coopérer et communiquer entre eux.

Nous supposons que l'environnement dans lequel les robots se déplacent est divisé en un ensemble de cellules. Chaque robot ne peut visionner que six emplacements ou cellules (voir figure 6.2.) et chaque cellule peut contenir un robot, un POI comme il peut être vide (voir figure 6.3)

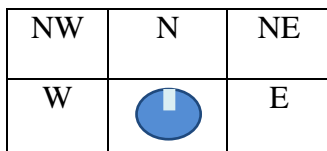


Figure 6.2. La vision locale de robot

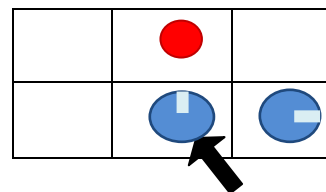


Figure 6.3. L'occupation des cellules qui entourent le robot

Dans notre travail, un robot exécutant l'algorithme d'exploration (décrit dans le chapitre 5) ne considère que les trois emplacements (N, W, E) dans sa vision locale. Tandis que la perception locale coopérative est déterminée par la portée de détection R_s du robot.

Chapitre 6 : Mise-en-œuvre et résultats de simulation

Comme il est montré dans le chapitre précédent (voir chapitre 5) la table de consultation concernant les règles sensorimotrices, exprimant la coopération des robots, est structurée comme suit :

<Perception locale coopérative> <état du robot>→<action>

Le premier bit de la table de consultation est réservé pour la perception locale. Deux bits sont réservés pour représenter l'état du robot et nous avons aussi réservés deux autres bits pour désigner l'action à effectuer.

Etant donné que le robot ne peut percevoir que l'un des cas suivants :

- *Au moins un* POI non encore couvert codé par **0**
- *Aucun* POI codé par **1**

Chaque robot peut être dans l'un des quatre états suivants :

1. **Bloqué** codé par **00**
2. **Attiré** code par **01**
3. **Libre** codé par **10**
4. **Trouvé** code par **11**

Le robot peut effectuer l'une des quatre actions suivantes :

- **Couvrir** codé par **00**
- **Explorer** codé par **01**
- **Aller à** codé par **10**
- **Ne rien faire** codé par **11**

La taille de la table de consultation coopérative égale donc à $2 \times 4 = 8$ entrées. Avec quatre actions possibles on se retrouve avec $4^8=65536$ tables. Chaque table à la structure suivante :

PL	Etat		Action	
0	0	0	0/1	0/1
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1	0/1	0/1

8 entrées

Figure 6.4. Structure de la table de consultation coopérative : 8 entrées dans chaque table

En suivant le même raisonnement que pour la coopération, la table de consultation concernant l'adaptation est structurée comme suit:

<Taux d'échecs de connexion> <protocole courant>→<Action>

Chapitre 6 : Mise-en-œuvre et résultats de simulation

Un seul bit est réservé pour exprimer le TLE, un autre bit pour désigner le protocole courant des robots et un troisième bit pour la représentation de l'action à exécuter.

Deux valeurs de taux local d'échec seront considérées :

1. **TLE <seuil** codé par **0**
2. **TLE >seuil** codé par **1**

Deux protocoles de communication peuvent être utilisés :

1. **AODV** codé par **0**
2. **Epidémique** codé par **1**

Etant donné que le robot ne peut exécuter que l'une de deux actions suivantes :

1. **Persister** codé par **0**
2. **Switcher** codé par **1**

La taille de la table de consultation concernant l'adaptation égale donc à $2 \times 2 = 4$ entrées. Avec deux actions possibles on se retrouve avec $2^4=16$ tables. Chaque table a la structure suivante :

TLE	Protocole	Action
0	0	0/1
0	1	...
1	0	...
1	1	0/1

4 entrées

Figure 6.5. Structure de la table de consultation concernant l'adaptation:4 entrées dans chaque table.

6.3.Résultats de simulation

La première phase consiste à trouver la meilleure particule qui optimise la fonction objective (**Eq6**) en utilisant la méthode d'optimisation PSO. Ensuite, la particule obtenue doit être interprétée en des règles plus significatives. La phase suivante consiste à prouver que les règles décelées sont insensibles aux changements du nombre de robots, au nombre des POI, à la taille de l'environnement, le rayon de détection et le rayon de communication utilisé, c'est la phase de la mise à l'échelle. Dans la dernière phase on compare les résultats obtenus via le routage adaptatif avec ceux obtenus par d'autres travaux.

Tous les résultats présentés sont programmés dans l'environnement de simulation basé agents « Netlogo ». La figure 6.6 montre le cas d'un robot bloqué qui communique avec ses voisins et la propagation des messages dans le réseau. La communication dans le cas de détection des POI non couverts est illustrée par la figure 6.7.

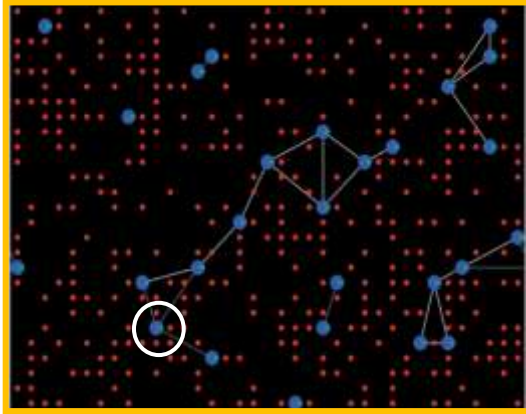


Figure 6.6. Diffusion du message par un robot bloqué : le robot bloqué est montré dans un cercle tandis que les lignes grises représentent le processus de communication

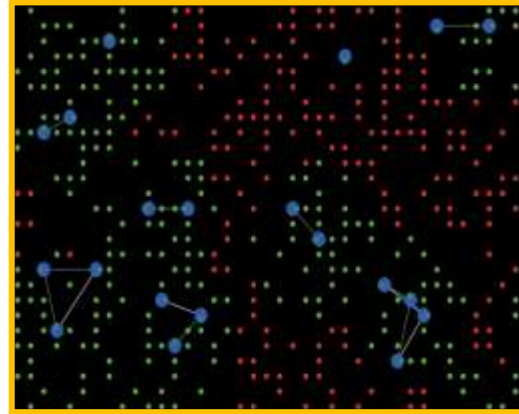


Figure 6.7. Diffusion de message lors de détection des POI non couverts : le disque vert est POI couvert tandis que le disque rouge représente POI non couvert

Les paramètres de l'algorithme de PSO sont résumés dans la table suivante (voir Tab6.1)

Paramètres de l'algorithme de PSO	
Nombre d'itérations	50
Nombre de populations	50
C1	0.12
C2	1.2
W	0.9

Tab6.1 Paramètres de l'algorithme de PSO

La figure 6.8 montre les résultats obtenus après les 50 itérations avec les paramètres de simulation déterminés dans la table suivante (voir Tab6.2) (on suppose que chaque patch simule vingt mètres en réalité)

Paramètres de simulation	
Taille d'environnement	400m x 400m
Nombre de robots	18
Nombre de POI	400
Portée de détection	60m
Portée de communication	100m
Nombre d'expérience	10
Pas de simulation	500
Seuil de TLE	0.75

Tab6.2 Paramètres de simulation

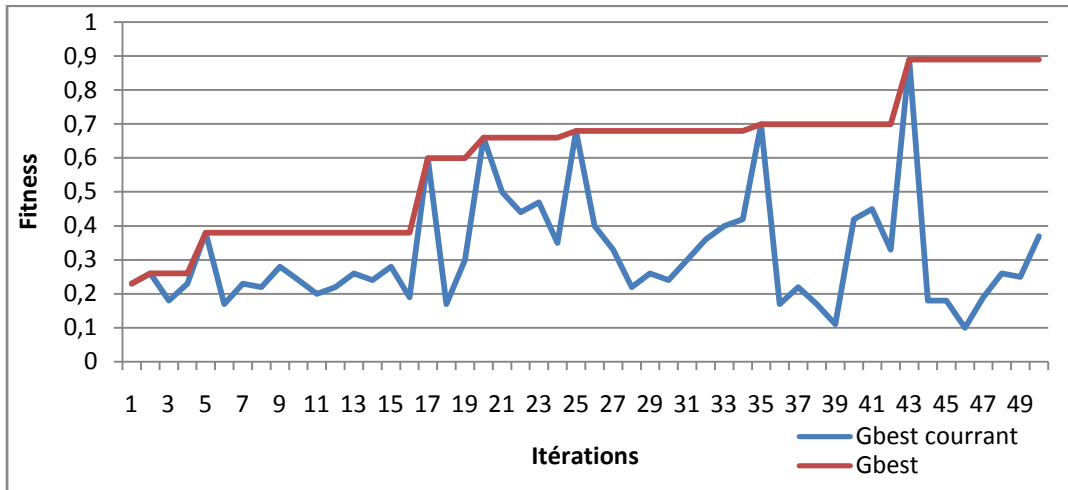


Figure 6.8. Valeurs de Gbest courant et Gbest dans les cinquante itérations

L'algorithme PSO trouve la meilleure valeur de fitness après une série d'itérations. Le gbest trouvé représente les deux meilleurs comportements. Le premier concerne le comportement de la coopération tandis que le second est le comportement d'adaptation.

On remarque que l'algorithme PSO arrive à trouver une fitness de meilleure valeur après une suite de perturbation dans les valeurs de Gbest courant. Autrement dit, l'algorithme de PSO cherche à maximiser la fonction objective (Eq6) en maximisant le taux de couverture des points d'intérêt (TCPI) et le taux de transmission de paquets avec succès (TTPS) comme il est illustré dans la figure 6.9 et figure 6.10 respectivement.

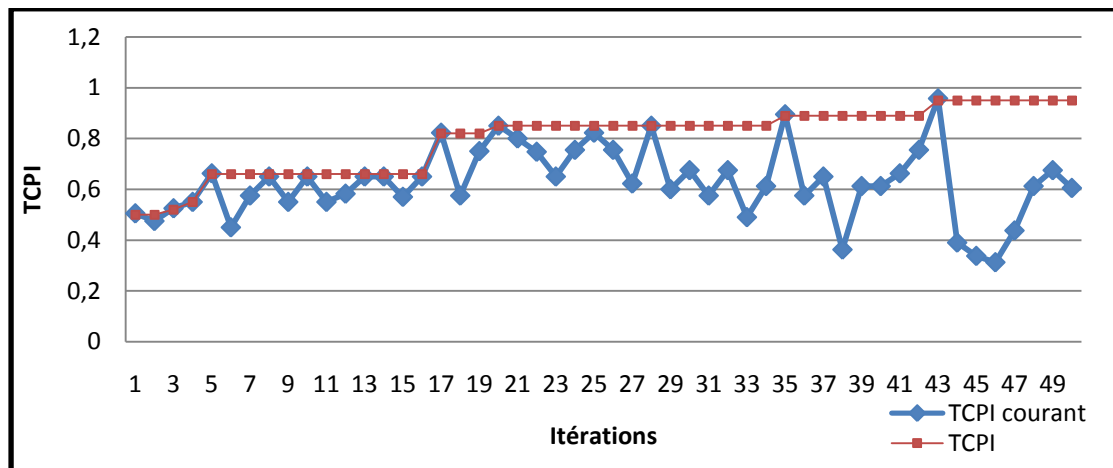


Figure 6.9. Valeurs de TCPI courant et TCPI dans les cinquante itérations

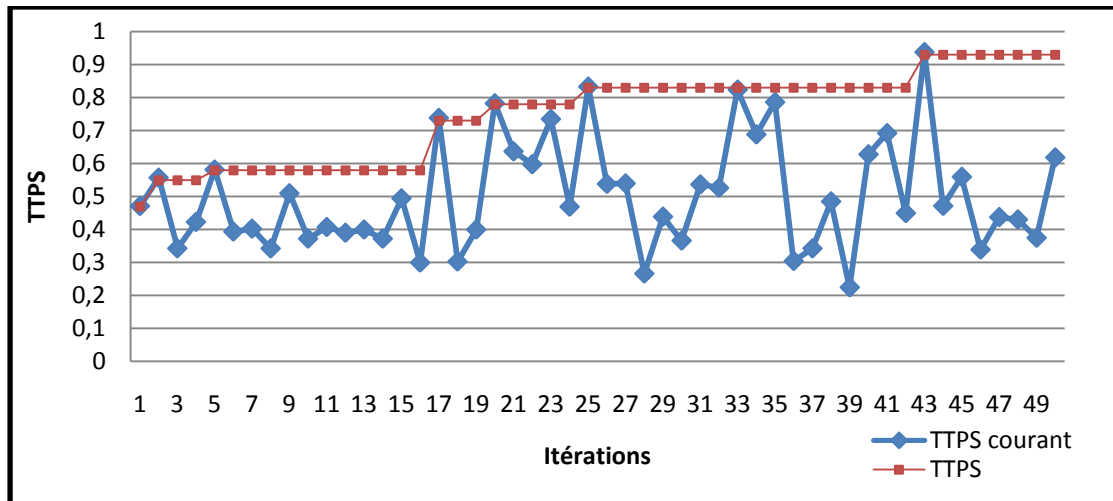


Figure 6.10. Valeurs de TTPS courant et TTPS dans les cinquante itérations

6.3.1. Les règles décelées

Après le déroulement de l'algorithme PSO, on arrive à tirer un ensemble de règles pour les deux comportements (coopération et adaptation). Les règles concernant la coopération sont les suivantes :

- Lorsque un robot détecte au moins un POI dans sa zone de détection, l'action de couverture sera toujours exécutée avec ou sans changement d'état en fonction de son état actuel.
- Quand un robot ne détecte pas de POI et son état est bloqué ou trouvé, alors il ne fait rien.
- Quand un robot ne détecte pas de POI et son état est attiré, il exécute l'action aller à.
- Lorsque un robot ne détecte pas de POI et son état est libre, alors il exécute l'action explorer.

Les règles que nous pouvons détecter pour le comportement d'adaptation sont:

- Si le taux local d'échec (TLE) est acceptable ($TLE < \text{seuil}$) et le protocole courant utilisé est l'AODV alors le robot doit maintenir l'utilisation de la stratégie de communication MANET.
- Si le TLE est acceptable et le protocole courant utilisé est l'épidémique, alors le robot doit switcher à la stratégie de communication MANET.
- Si le TLE n'est pas acceptable ($TLE > \text{seuil}$) et le protocole courant utilisé est AODV, alors le robot doit switcher à la stratégie de communication DTN.
- Si le TLE n'est pas acceptable et le protocole courant utilisé est l'épidémique, alors le robot doit maintenir l'utilisation de la stratégie de communication DTN.

Chapitre 6 : Mise-en-œuvre et résultats de simulation

Les règles d'adaptation sont résumées dans la figure suivante (voir figure 6.11)

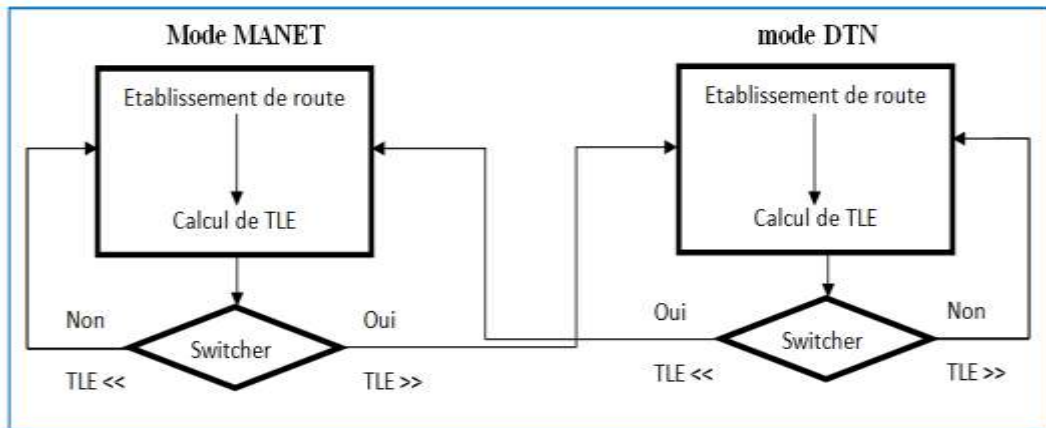


Figure 6.11. Mécanisme d'adaptation

6.3.2. La phase de la mise à l'échelle

L'étape de mise à l'échelle est appliquée sur les règles que nous avons détectées afin de prouver qu'elles sont insensibles aux changements des paramètres tels que le nombre de robots (figure 6.12), le nombre de POI (figure 6.13), la taille de l'environnement (figure 6.14), la portée de détection (figure 6.15) et la portée de communication (figure 6.16).

6.3.2.1. La mise à l'échelle du nombre de robots

Nous varions le nombre de robots de 5 à 75 robots et nous mesurons le temps d'exploration. La figure 6.12 montre que le temps nécessaire pour explorer les POI diminue lorsque le nombre de robots augmente. Ceci s'explique par l'exécution parallèle de la tâche d'exploration, le comportement de la coopération des robots et la diminution de taux de déconnexion.

On remarque que lorsque le réseau devient plus dense (nombre de robots > 65 robots), l'exploration commence à prendre plus de temps pour finir. Ceci peut être expliqué par le fait que le nombre important de robots influe négativement sur le temps d'exploration, en d'autres termes dans le cas de collision des robots, ces derniers gênent les uns les autres.

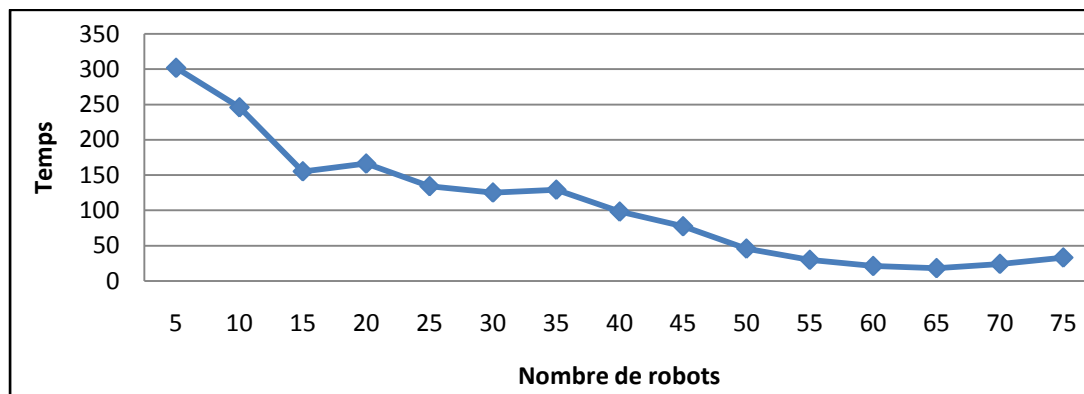


Figure 6.12. Mise à l'échelle du nombre de robots

6.3.2.2. La mise à l'échelle du nombre de points d'intérêt

La figure 6.13 montre la relation entre la variation du nombre de points d'intérêt et le temps d'exploration. Contrairement à la figure 6.12, le temps d'exploration augmente selon l'augmentation du nombre de POI; un plus grand nombre de points d'intérêts dispersés prend plus de temps pour être exploré.

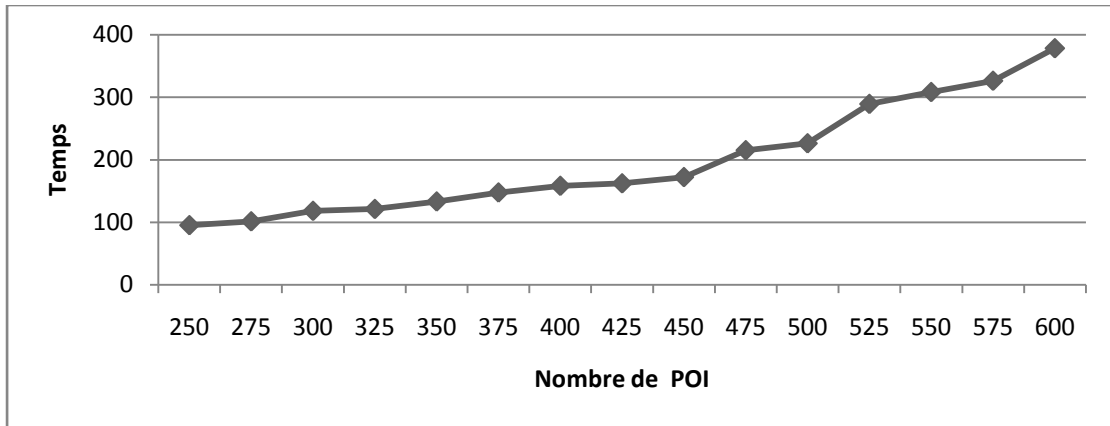


Figure 6.13. Mise à l'échelle du nombre de POI

6.3.2.3. La mise à l'échelle de la taille d'environnement

Le troisième cas de mise à l'échelle concerne les temps d'exploration obtenus en fonction des changements dans la taille de l'environnement (voir la figure 6.14). Nous notons que le temps d'exploration augmente progressivement avec l'augmentation de la taille de l'environnement. Les robots prennent plus de temps pour trouver un ensemble de POI répartis dans un vaste environnement. Cependant, une grande surface d'environnement réduit la probabilité de collision des robots et permet aux robots de se déplacer avec plus de liberté.

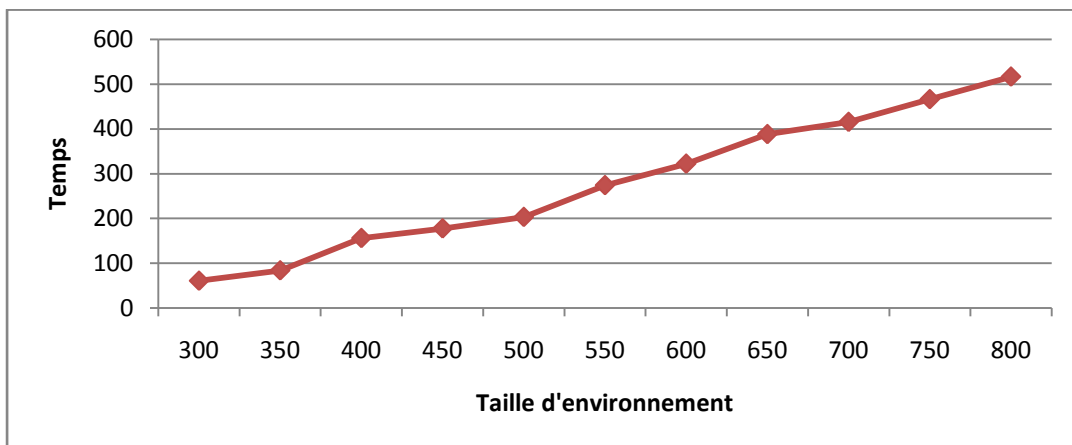


Figure 6.14. Mise à l'échelle de la taille d'environnement

6.3.2.4. La mise à l'échelle du rayon de détection

On applique des changements seulement dans la portée de détection des robots afin de mesurer le temps d'exploration. Les résultats obtenus (voir figure 6.15) montrent la

Chapitre 6 : Mise-en-œuvre et résultats de simulation

relation inversement proportionnelle qui existante entre la portée de détection et le temps d'exploration ; plus que la portée de détection est large et plus les robots explorent les POI plus rapidement. Avec une large portée de détection le robot peut détecter un nombre important des POI, cependant la largeur de portée de détection est limitée par la capacité de détection des capteurs des robots.

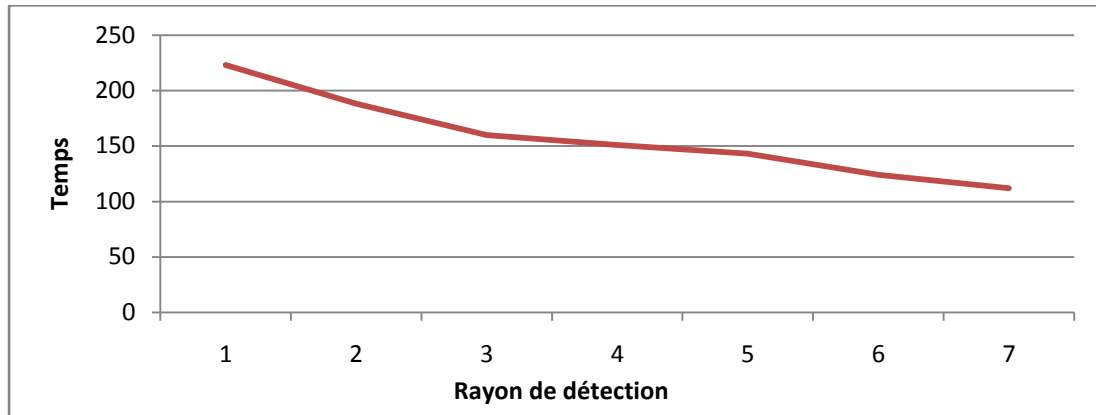


Figure 6.15. Mise à l'échelle du rayon de détection

Afin d'étudier l'influence de la taille d'environnement sur le taux de couverture des points d'intérêt (TCPI), les tests suivants sont effectués. A chaque fois, nous fixons la portée de détection des robots et nous mesurons les TCPI obtenus dans différentes tailles d'environnement (pour 400 POI et 18 robots). Le critère d'arrêt dans cette expérience est le meilleur temps d'exploration trouvé au-dessus.

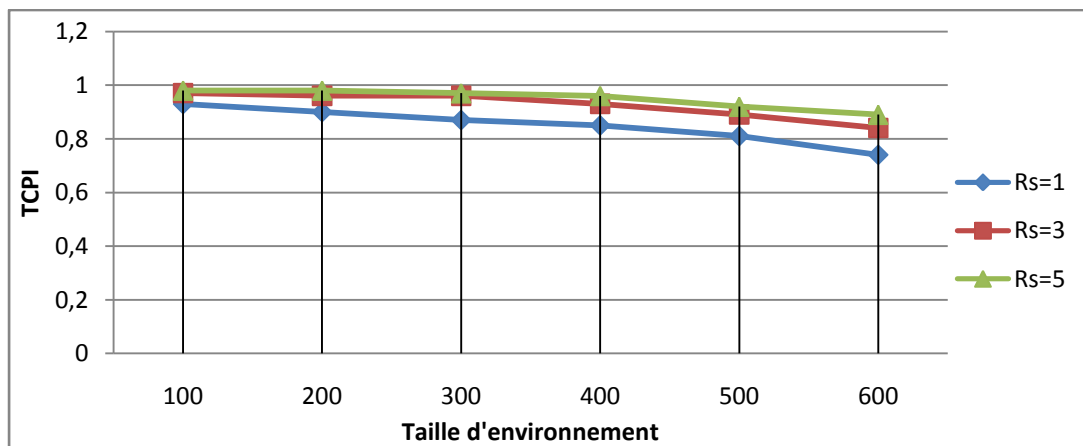


Figure 6.16. Le taux de couverture des points d'intérêt en fonction de la taille de l'environnement et le rayon de détection

On observe que plus la taille d'environnement augmente plus le TCPI diminue même avec une large portée de détection. Avec un environnement de petite taille, les trois portées de détection les robots arrivent rapidement à détecter un nombre important des POI. Les POI dispersés dans une grande surface prennent plus de temps pour être explorés même avec des portées de détection raisonnable ($R_s=1$ et $R_s=5$). Cependant,

Chapitre 6 : Mise-en-œuvre et résultats de simulation

un rayon de détection ($R_s=5$) permet toujours d'augmenter le TCPI par rapport aux deux autres rayons même avec l'augmentation de la taille d'environnement.

6.3.2.5. La mise à l'échelle du rayon de communication

Dans cette expérience on fixe tous les paramètres (18 robots et 400 POI) et on fait varier le rayon de communication (R_c) des robots. Comme il est montré dans la figure 6.17 le temps nécessaire pour explorer l'ensemble des POI diminue en fonction de l'augmentation du rayon de communication. L'augmentation de R_c engendre la détection de plus de voisins d'un seul saut, donc la propagation des messages sera faite plus rapidement et par conséquent une coopération efficace offre un meilleur temps d'exploration.

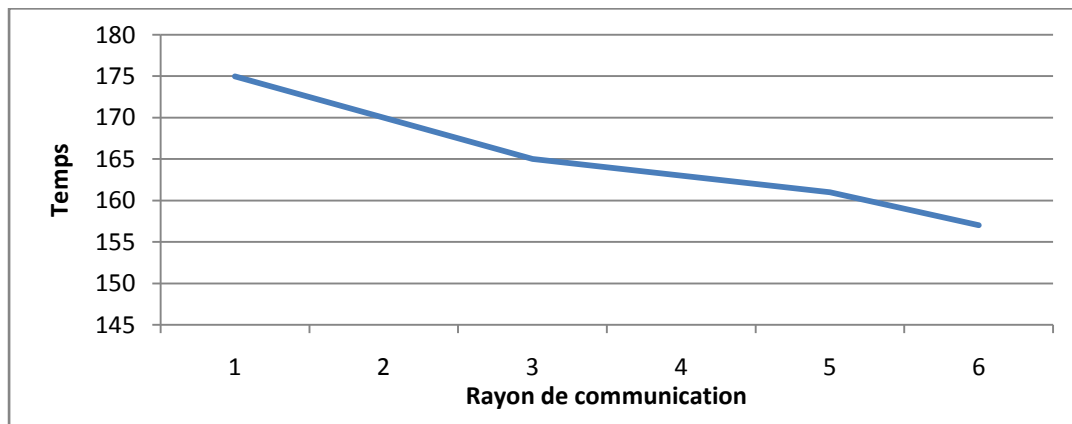


Figure 6.17. Mise à l'échelle du rayon de communication

La figure 6.18 montre la relation entre le rayon de communication (R_c) et le taux de transmission de paquets avec succès (TTPS) et l'influence du nombre de robots sur cette relation.

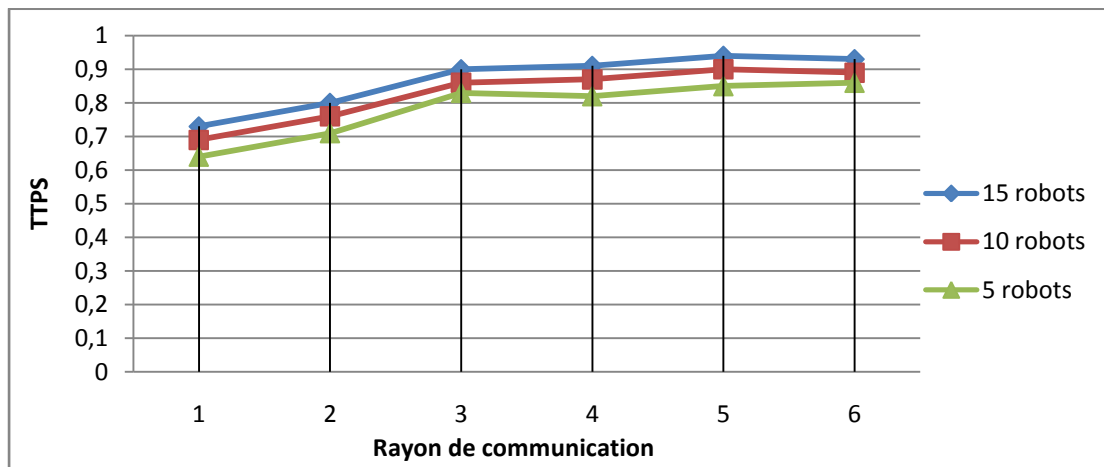


Figure 6.18. Le taux de transmission de paquets avec succès en fonction du rayon de détection

On remarque que le rayon de communication influe sur le TTPS ; plus que le rayon de communication augmente plus que le taux TTPS augmente. On explique cette relation par le raisonnement suivant : un grand rayon de communication engendre plus de

Chapitre 6 : Mise-en-œuvre et résultats de simulation

voisins dans un seul saut, donc le message envoyé par un robot sera reçu par d'autres robots (ses voisins), par conséquent le nombre de messages reçus augmente et le taux TTPS augmente.

Avec un petit nombre de robots dispersés dans un large environnement on trouve que la probabilité de communication diminue considérablement et par conséquent le nombre de messages perdus augmente ce qui minimise le TTPS. Cependant, si le rayon de communication est assez grand (avec 5 robots et $R_c = 6$) on arrive à un TTPS élevé. On remarque que plus le réseau devient plus dense plus les robots peuvent transmettre la plupart des messages avec succès même avec un rayon de communication très restreint.

6.4. Comparaison des résultats

Afin de valider les performances de notre routage adaptatif, nous avons fait des comparaisons avec d'autres travaux similaires en se basant sur les critères de comparaison suivants:

1. Temps d'exploration
2. Le TTSP
3. Le TCPI

Nous avons comparé notre routage adaptatif avec le travail de Tuan Le et al. [60]. Ils ont considéré le problème d'exploration qui utilise la prise de conscience de la connectivité « connectivity-awareness » avec un control distribué de mouvement dans un système multi-robots pour diminuer le temps d'exploration. Le travail consiste à maintenir la connectivité entre les robots pour accomplir une tâche d'exploration en utilisant les réseaux Manet comme stratégie de communication. Dans ce cas, la métrique de comparaison est l'efficacité d'exploration avec une portée de communication limitée.

Comme il est montré dans la figure 6.19, l'exploration avec le routage adaptatif prend moins de temps pour se terminer par rapport à l'exploration proposée dans [60]. Donner plus de liberté à la mobilité des nœuds en utilisant le routage adaptatif pour la communication augmente la coopération entre les robots et par conséquent ceci diminue le temps d'exploration.

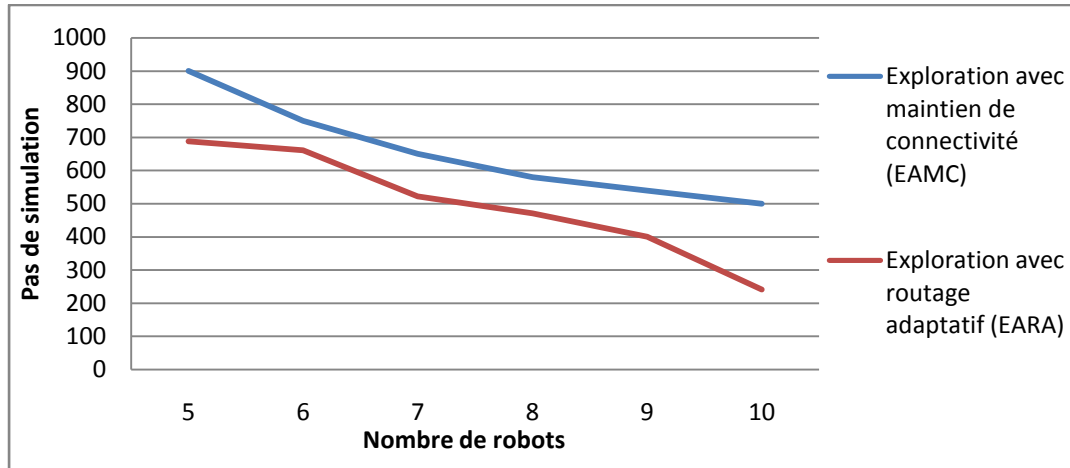


Figure.6.19.Comparaison entre EAMC et EARA par rapport au temps d'exploration en fonction du nombre de robots

Nous avons comparé aussi les résultats obtenus via le routage adaptatif avec ceux obtenus en utilisant seulement la stratégie de communication Manet et cette fois sans appliquer aucune méthode pour maintenir la connectivité entre les robots. En d'autres termes, nous avons gardé le même scénario pour la tâche d'exploration, mais nous avons modifié le mécanisme de communication; les robots utilisent seulement le protocole AODV pour leurs communications même avec des déconnexions fréquentes.

Les résultats obtenus (voir figure.6.20) montrent que le temps écoulé pour explorer les POI avec un routage MANET est très grand par rapport à un routage adaptatif. Ceci s'explique par le fait que le mécanisme d'adaptation joue un rôle très important pour augmenter le TTPS ce qui renforce la coopération et par conséquent accélère la tâche d'exploration.

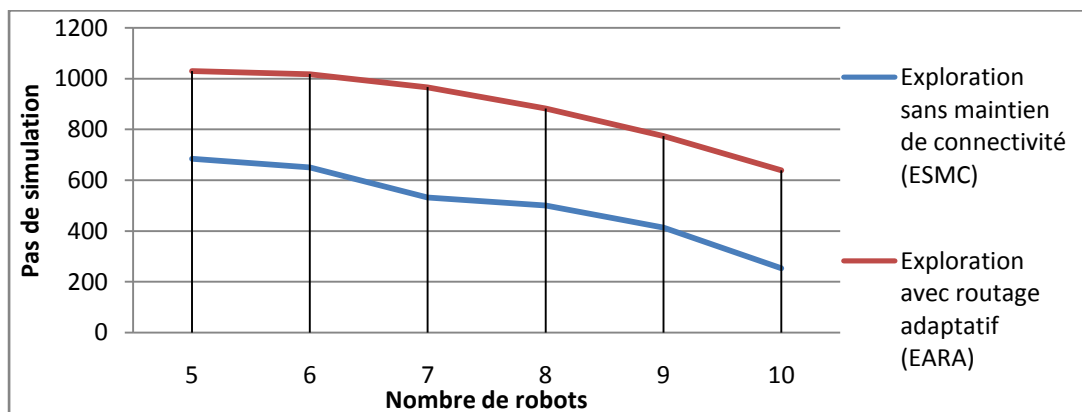


Figure.6.20.Comparaison entre ESMC et EARA par rapport au temps d'exploration en fonction du nombre de robots

Une comparaison des trois stratégies d'exploration ; exploration avec maintien de connectivité [60] (EAMC), exploration avec routage adaptatif (EARA) et exploration sans maintien de connectivité (ESMC) est illustrée dans la figure 6.21.

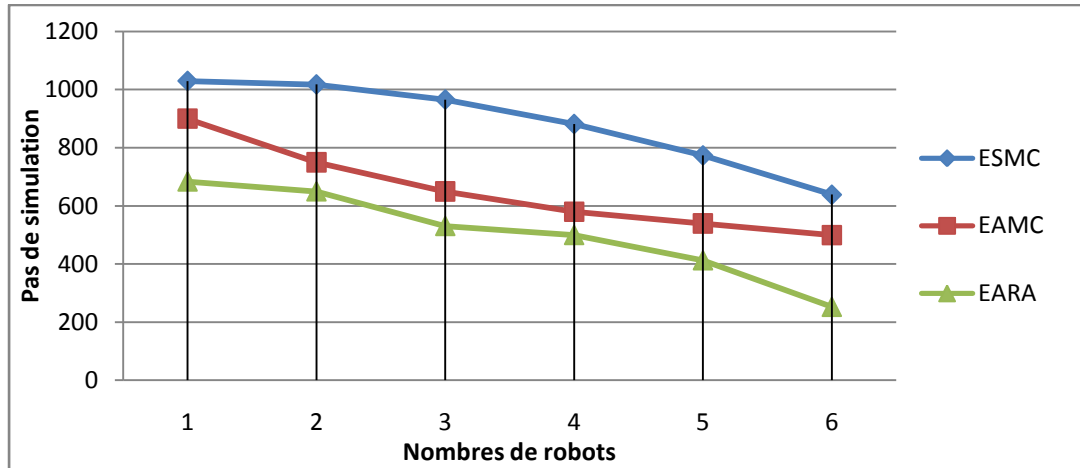


Figure 6.21. Comparaison entre ESMC, EAMC et EARA par rapport au temps d'exploration en fonction de nombre de robots

On remarque que plus la densité du réseau augmente plus que les deux stratégies d'exploration (avec et sans maintien de connectivité) donnent des résultats plus proches. Plus que le nombre de robots augmente plus que le réseau devient connecté, et avec une grande densité les stratégies de maintien de connectivité ne sont plus nécessaires pour garantir la transmission des messages. La figure 6.22 présente le taux de couverture des points d'intérêt (TCPI) en fonction de la taille de l'environnement avec les deux stratégies d'exploration ; EARA et ESMC. Les résultats obtenus montrent que le TCPI diminue avec l'augmentation de la taille d'environnement. Dans un environnement de petite taille l'ESMC donne un TCPI élevé car les robots sont proches ce qui augmente la probabilité de transmission des messages et permet la coopération entre ces robots. Cependant, le TCPI diminue considérablement en fonction de l'augmentation de la taille d'environnement, due au problème de connectivité entre les robots ce qui rend la coopération plus difficile. On observe que l'EARA apporte une amélioration de TCPI dans un environnement de petite taille. Cette amélioration diminue un peu, mais elle reste supérieure à l'ESMC pour un environnement assez grand. Ceci peut être expliqué par le fait que les robots peuvent communiquer et coopérer même s'ils sont dispersés grâce au mécanisme d'adaptation de la stratégie de communication.

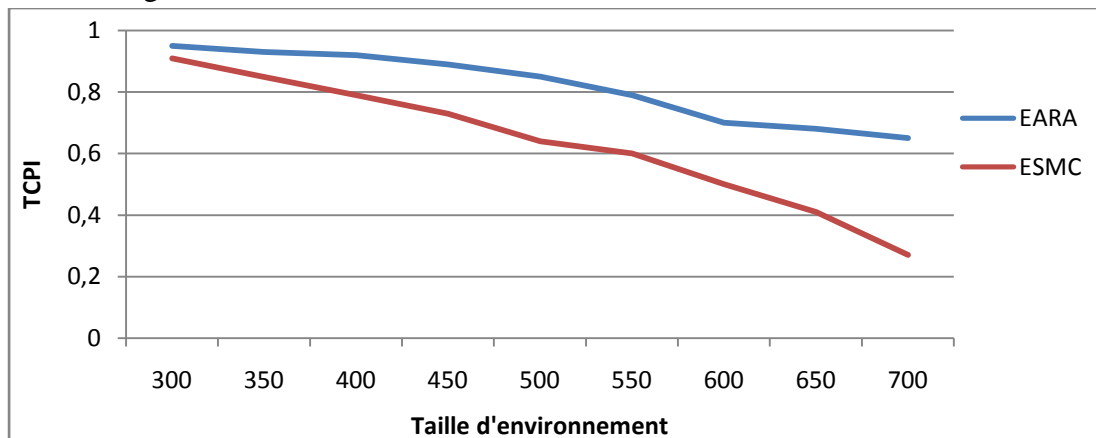


Figure 6.22. Comparaison entre ESMC et EARA par rapport au TCPI en fonction de la taille d'environnement

Chapitre 6 : Mise-en-œuvre et résultats de simulation

La figure 6.23 présente le taux de transmission des paquets avec succès (TTPS) en fonction du nombre des robots pour les deux stratégies d'exploration ESMC et EARA. On constate d'après les résultats obtenus que l'EARA apporte des améliorations considérables de TTPS par rapport à l'ESMC pour un réseau composé de moins de 15 robots. On peut expliquer ce phénomène par l'utilisation du routage épidémique qui augmente la probabilité de transmission des messages ce qui n'est pas le cas avec un routage Manet.

Les TTPS augmentent proportionnellement avec la densité du réseau pour l'ESMC, mais ils sont toujours plus petits que ceux de l'EARA. Pour un réseau composé de 20 robots les TTPS des deux stratégies d'exploration commencent à être très proches. Donc la densité du réseau renforce la livraison des messages.

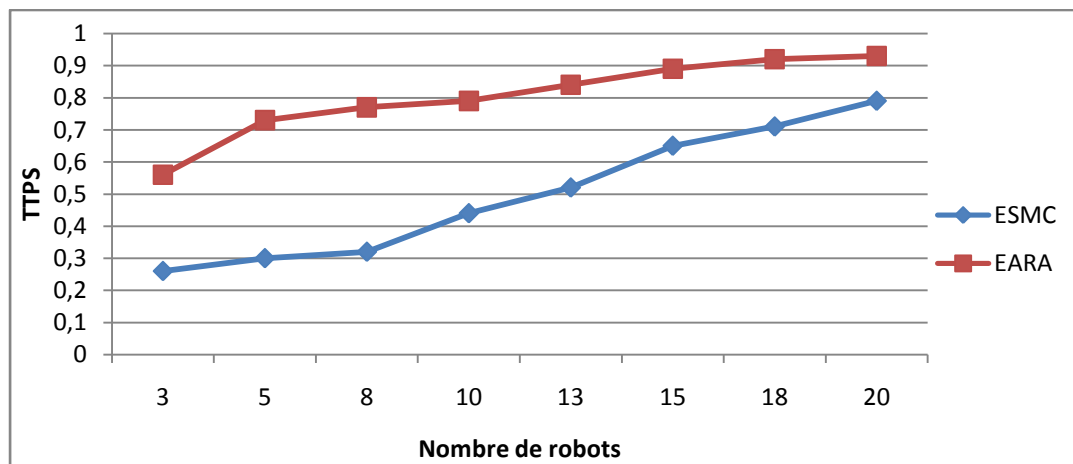


Figure 6.23. Comparaison entre ESMC et EARA par rapport au TTPS en fonction de nombre de robots

6.5.Conclusion

Dans ce chapitre nous avons expliqué l'utilité de la simulation pour reproduire les hypothèses et les méthodes proposées. Afin de concrétiser nos propositions, le simulateur Netlogo est choisi. Nous avons présenté notre système ainsi que les caractéristiques des robots constituant ce système. Les différents états ainsi que les actions effectuées par le réseau de robots sont décrits.

Nous avons présenté les résultats obtenus, les règles décelées ainsi que l'étape de la mise à l'échelle qui démontre que les règles détectées sont insensibles aux variations des paramètres les plus importants tels que le nombre des robots, de la taille d'environnement et du nombre de points d'intérêt. Nous avons finalisé ce chapitre par la comparaison du routage adaptatif proposé dans ce mémoire avec d'autres travaux. Les résultats obtenus après la comparaison montrent que le routage adaptatif, que nous proposons, l'emporte sur les autres protocoles de tous points de vue. En outre, il arrive à accélérer la tâche d'exploration significativement par rapport à ces protocoles.

Conclusion générale et Perspectives

L'utilisation d'une collectivité de robots connectés, dans une mission d'exploration, élargie l'espace des possibilités pour solutionner ce problème. Par contre, ceci engendre un nouveau problème. De par sa mobilité, la connectivité du réseau de robots peut être intermittente. Ce qui aura forcément des effets néfastes sur le processus de communication entre les différents éléments du réseau de robots considéré, qui doivent coopérer pour venir à bout de leur mission. Ceci rend les protocoles de routage, aussi bien ceux adoptés par les MANETs que ceux adoptés par les DTNs inefficaces dans le cas de notre réseau de robots. Si les coupures sont fréquentes, un protocole comme AODV ne pourra pas résister à cette situation et il deviendrait très vite inopérant. Une solution possible dans un tel cas consiste en la mémorisation des informations recueillies et leur transmission quand cela est possible. Ceci nous fait transiter d'une politique classique d'acheminement de l'information vers une politique du store and forward appliquée dans les DTN. Cependant, avec un nombre important de points d'intérêts et une capacité de mémorisation faible la technique appliquée dans les DTN serait inefficace.

La solution proposée, dans cette perspective, est d'utiliser un protocole de communication adaptatif permettant de transmettre les paquets lorsque les conditions de transmissions classiques sont vérifiées et de les stocker en vue de les diffuser à la première occasion de connexion qui se présente lorsque ces conditions ne sont pas vérifiées.

Dans ce mémoire nous nous sommes intéressés au problème de déconnexion dans un réseau de robots. Au début, nous avons étudié les réseaux de robots de manière générale ainsi que les réseaux sans fil et nous avons déterminé les problèmes d'exploration et de couverture. Par la suite, nous avons présenté les méthodes d'optimisation en générale et l'optimisation par essaim particulaires (PSO) en particulière.

Une fois le problème posé, nous avons proposé un routage adaptatif qui se présente, alors, comme la solution la plus appropriée à ce genre de situations. Il s'agit d'un routage qui commute entre une solution de type MANET et une autre de type DTN selon le cas qui se présente devant lui et ceci pour permettre à la mission d'exploration qui incombe au groupe de robots en question de se voir accomplir dans les meilleures conditions de communications possibles. Autrement dit, le choix du protocole de routage à adopter dépend de la connectivité. Un protocole inspiré des protocoles appliqués dans les réseaux DTN sera adopté en cas où on aura une connexion intermittente sinon un protocole inspiré des protocoles appliqués dans les MANET sera suffisant.

Conclusion générale et perspectives

Nous avons simulé notre proposition et nous avons pu tirer un ensemble de règles qui reflètent le comportement de coopération et d'adaptation des robots. Une phase de mise à l'échelle a été faite pour prouver la consistance des règles décelées au changement des paramètres de simulation. La dernière phase consistée à comparer notre routage adaptatif avec des travaux similaires. Sur ceux, notre routage adaptatif a montré son efficacité, il est arrivé à minimiser le temps d'exploration tout en assurant la transmission d'un nombre important de messages même avec des déconnexions fréquentes.

Comme suite à ce travail, nous voulons entreprendre une étude comparative, via la simulation, entre une solution de routage adaptatif, s'appuyant sur des protocoles de type MANETs et DTNs et une solution de routage adaptatif bio-inspiré, qu'on doit proposer dans les deux cas, en vue de réussir aussi bien la couverture que l'exploration dans le contexte d'un réseau de robot. Le succès d'un tel travail, peut avoir des répercussions sans précédent dans des domaines aussi stratégiques que l'industrie, la sécurité et la santé. Nous envisageons tout ça dans nos futurs travaux de recherche qu'ils soient d'ordre individuel, professionnels ou académique.

Références bibliographiques

- [1] J. Christophe, B. MAISONNIER, G.GÂTÉ, P. DE PAS, P. FABIANI, La robotique, La jaune et la rouge, La revue de la communauté polytechnicienne N ° 655, Mai 2010.
- [2] A-R. Mosteo, L. Montano, and M-G. Lagoudakis, « Multi-Robot Routing under Limited Communication Range », 2008 IEEE International Conference on Robotics and Automation. Pasadena, CA, USA, May 19-23, 2008.
- [3] K-M. Wurm, C. Stachniss and W.Burgard, « Coordinated Multi-Robot Exploration using a Segmentation of the Environment », 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems Acropolis Convention Center Nice, France, Sept, 22-26, 2008.
- [4] W. Burgard M. Moors, C. Stachniss and F-E. Schneider, « Coordinated Multi-Robot Exploration », IEEE TRANSACTIONS ON ROBOTICS, VOL. 21, NO. 3, JUNE 2005 .
- [5] S. KAFHALI, A.HAQIQ, « Effect of Mobility and Traffic Models on the Energy Consumption in MANET Routing Protocols » International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-3, Issue-1, March 2013.
- [6] S-C. MIN, H-C. JHAN, H-L JHIH, S. GAO, « Routing and Buffering Strategies in Delay-Tolerant Networks: Survey and Evaluation » International Conference on Parallel Processing Workshops, IEEE 2011.
- [7] Krishnanand KR, Nayak SK, Panigrahi BK, Rout PK (2009) Comparative study of five bio-inspired evolutionary optimization techniques. In: Abraham A, Herrera F, Carvalho A, Pai V (eds) 2009 world congress on nature and biologically inspired computing (NABIC 2009), pp 1230–1235.
- [8] R-V. Kulkarni, and G-K Venayagamoorthy, « Technical Correspondence: Particle Swarm Optimization in Wireless-Sensor Networks: A Brief Survey », IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS, VOL. 41, NO. 2, MARCH 2011.
- [9] L. Lhotská, M. Macaš, and M. Burša, « PSO and ACO in Optimization Problems », Springer, pp. 1390-1398, 2006.
- [10] C. Pinciroli, R. O'Grady, A-L. Christensen, M. Birattari, M.Dorigo, « Parallel Formation of Differently Sized Groups in a Robotic Swarm », SICE Journal of Control, Measurement, and System Integration, volume 52, number 3, pages 213-226, 2013.
- [11] S.Haddad, J. El Haddad, "Algorithmes de communication auto-stabilisants dans un système de robots mobiles", 4e Colloque Francophone sur la Modélisation des Systèmes Réactifs - MSR'03, 2003.
- [12] E. Bonabeau, M. Dorigo, G. Theraulaz, "Swarm Intelligence: From Natural to Artificial Systems", New York, NY: Oxford University Press, 1999.
- [13] J. C. Barca and Y. A. Sekercioglu, "Swarm robotics reviewed," Robotica, vol. FirstView, pp. 1-15, 2012.
- [14] V. Kumar, D. Rus, G- S. Sukhatme, « Networked robots », Springer Handbook of Robotics Part E page pp 943-958, 2008.
- [15] M. Zagane, "une architecture décisionnelle de contrôle pour un groupe de robots mobiles coopératifs dans un environnement dynamique et non structuré", mémoire en vue de l'obtention de magister en informatique, Ecole nationale supérieure d'informatique, 2010.
- [16] C. Ronald KUBE, Hong ZHANG – Collective Robotics Intelligence: From Social Insects to Robots – Proceedings, 3rd International Conference on Simulation of Adaptive Behavior, vol. 2, pp.189 -218 1994.
- [17] S. Hoceini, A. Mellouk, A. Chibani, Y. Touati et B. Augustin , « Swarm intelligence routing approach in networked robots », annals of telecommunications-Annales des telecommunications Volume 67, Issue 7-8 , pp 377-386, 2012.
- [18] J. Beaudry, "Système multi-robots autonome et coopératif", article de synthèse ELE6904 SÉMINAIRES, Avril 2004.

Références bibliographiques

- [19] C. NEHNOUH, F. DEBBAT, « Robotique Collective Implémentation d'un comportement d'agrégation stratégique inspiré de comportement des Cafards », Université Mustapha Stambouli – Mascara, 2006.
- [20] R. Drouilly, "SLAM Visuel 3D pour robot mobile autonome", Projet de Fin d'étude, université de Strasbourg, 2011.
- [21] L. Adouane, "architecture de contrôle comportementales et réactives pour la coopération d'un groupe de robots mobiles", thèse en vue de l'obtention du doctorat de l'université de Toulouse, 2005.
- [22] V. BRAITENBERG. *Vehicles: Experiments in synthetic psychology*, MA : MIT Press, Cambridge, 1984.
- [23] Y. Yan, T.Zhenmin, "Control Architecture for autonomous multi-robot system: Survey and Analysis", *Intelligent Computation Technology and Automation*, 2009. ICICTA '09. Second International Conference on (Volume:4), pages 376 - 379, 2009.
- [24] Brian P. Gerkey and Maja J MatariC. Sold: Auction methods for multi-robot coordination. *IEEE Transactions on Robotics and Automation*, 18(5):758-768, October 2002.
- [25] Z. YAN, "Contributions à la coordination de tâches et de mouvements pour un système multi-robots", thèse pour l'obtention du Doctorat de l'Université Paris 8 spécialité Informatique, 2012.
- [26] J. Christophe, B. MAISONNIER, G.GÂTÉ, P. DE PAS, P. FABIANI, « La robotique », *La jaune et la rouge, La revue de la communauté polytechnicienne* N ° 655, Mai 2010.
- [27] Z. WANG, L LIU and M. ZHOU, « Protocols and Applications of Ad-hoc RobotWireless Communication Networks: An Overview », *INTERNATIONAL JOURNAL OF INTELLIGENT CONTROL AND SYSTEMS* VOL. 10, NO. 4, DECEMBER 2005, 296-303.
- [28] F. Saïdi, G. Pradel, "Un cadre pour un système multi-robot de service", *APII-JESA*, 2005.
- [29] W. Shen, B. Salemi, P. Will, "Hormone-Inspired Adaptive communication and Distributed Control for CONRO Self- Reconfigurable Robots", *IEEE transactions on robotics and automation*, 2002
- [30] H. Jones, S. Rock, "Dialogue-Based Human-Robot Interaction for Space Construction Team", *Aerospace Conference Proceedings*, 2002. IEEE , pages 7-3645 - 7-3653 vol.7
- [31] T. Fong, C. Kunz, "The Human-Robot Interaction Operating System", *1st ACM SIGCHI/SIGART conference on Human-robot interaction*, 2006
- [32] C. BOURJOIT, « Système multi-agents : Modélisation et simulation informatique de comportement collectifs, Chapitre II : Les différents concepts et composants d'un SMA », *Master M2 Sciences Cognitives, Université de Nancy2*, 2008
- [33] D. David, « Théorie des SMA », *cours M1 STIC*, 2010
- [34] Kumar, V., G. Bekey, and A. Sanderson, "Networked Robots," *WTEC Panel on INTERNATIONAL ASSESSMENT OF RESEARCH AND DEVELOPMENT IN ROBOTICS DRAFT REPORT*, pages 57-72 2005
- [35] M. Schwager, J. McLurkin, D. Rus, "Distributed coverage control with sensory feedback for networked robots". *Robotics Science and Systems Conference*, 2009
- [36] P. Mandal, R-K Barai1, M. Maitra et S. Roy « From Theory to Practice: Collaborative Coverage Using Swarm Networked Robot in Localisation », S. Unnikrishnan, S. Surve, and D. Bhoir (Eds.): *ICAC3 2013, CCIS 361*, pp. 626–635, 2013
- [37] A. Mellouk, Y. Amirat, N. Hagita et D. Song, « Ubiquitous networked robots », *Ann. Telecommun.* (2012) 67:295–296, 2013
- [38] J. Reich et E. Sklar, « Robot-Sensor Networks for Search and Rescue », *Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics (SSRR)*, 2006.
- [39] CH. BISSERIEUX, P. LAURENT, M. CASTRO, « Les réseaux sans fil de proximité », *rapport, Institut national de recherche et de sécurité pour la prévention des accidents du travail et des maladies professionnelles*, 2012
- [40] E. Ben HAMIDA, « Modélisation stochastique et simulation des réseaux sans fil multi-sauts », *Thèse pour l'obtention du grade de docteur en informatique, l'institut National des Sciences Appliquées de Lyon*, 2009
- [41] S. KASSAB ; M. OULARBI , « élaboration d'un protocole de routage efficace en énergie pour les réseaux de capteurs sans fil », *Mémoire de fin d'études pour l'obtention du diplôme d'ingénieur d'état en informatique, école nationale supérieure d'informatique (ESI)*, 2010

Références bibliographiques

- [42] M. VAN DER MEERSCHEN, « Hybridation entre les modes ad-hoc et infrastructure dans les réseaux de type Wi-Fi », Mémoire de fin d'études présenté en vue de l'obtention du grade d'Ingénieur Civil Informaticien en Sciences Appliquées, Université Libre de Bruxelles, 2006
- [43] BS. HAGGAR, « Les protocoles de routage dans ad hoc », Rapport de stage, Université de Reims, 2007
- [44] S. KAMMOUN, « Implémentation de la QoS sur un protocole de routage (multicast) Ad hoc », Rapport de Projet de fin d'études, Ecole supérieure de communication de Tunis, 2006
- [45] L. BARRERE « Étude et proposition de services dans les réseaux mobiles militaires de type MANet », thèse pour obtenir le grade de docteur spécialité : informatique, université BORDEAUX I, 2009
- [46] A. BENMESSAOUD « Classification des protocoles de routage dans les Réseaux Tolérants aux Délais (DTN) », Mémoire En vue de l'obtention du diplôme de Magister En informatique Option : Ingénierie des Systèmes Informatiques ISI, 2009
- [47] C. MASCOLO, M. MUSOLESI, « A Community Based Mobility Model for Ad Hoc Network Research », the 2nd international workshop on Multi-hop ad hoc networks: from theory to reality, pages 31-38, 2006.
- [48] A. Khalil, R. Beghdad, « Coverage and Connectivity Protocol for Wireless Sensor Networks », 24th International Conference on Microelectronics (ICM), 2012.
- [49] K. DRIRA, « Topologie Dynamique Virtuelle Pour Le Routage Dans Les Réseaux Mobiles Ad hoc » Rapport de projet de fin d'études, école supérieure des communications de TUNIS, 2005
- [50] Z. WANG, L. LIU and M. ZHOU, « Protocols and Applications of Ad-hoc RobotWirelessCommunicationNetworks: An Overview », international journal of intelligent control and systems vol. 10, no. 4, december 2005, 296-303, 2006
- [51] A.K GUPTA, H. SADAWARTI, A.K. VERMA , « implementation of DYMO routing protocol », International Journal of Information Technology, Modeling and Computing (IJITMC) Vol.1, No.2, May 2013
- [52] S. ABID, M. SHAFI, « Energy efficient routing in mobile ad-hoc networks for healthcare environments », IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 1, No 1, January 2013
- [53] <http://www.inf.cyut.edu.tw/AIT2007/210.pdf>
- [54] A. ZEBDI, « DZ-MAODV : nouveau protocole de routage multicast pour les réseaux Ad hoc mobiles basé sur les zones denses », mémoire présenté à l'université du QUEBEC A TROIS-RIVIERES comme exigence partielle du programme de maîtrise en mathématiques et informatique appliquées, avril 2006.
- [55] A. ADOLF, « Conception d'un protocole de routage réactif sécurisé à l'aide de processeurs sécurisé embarqués pour les réseaux ad hoc », master recherche, université de Limoges, 2007
- [56] D. ESPECE, « Protocoles de routage réactifs pour l'optimisation de bande passante et la garantie de délai dans les réseaux ad hoc mobiles », thèse en vue de l'obtention du doctorat en informatique, université de TOULOUSE, 27 novembre 2008
- [57] Arnaud Casteigts, Amiya Nayak and Ivan Stojmenovic « Topology control in sensor, actuator and mobile robot Networks », Wireless Sensor and Actuator Networks, pages 185, 2010.
- [58] Basu, P. ;Redi, J. Movement control algorithms for realization of fault-tolerant ad hoc robot networks, Network, IEEE (Volume:18 , Issue: 4), 2004.
- [59] M. M. Zavlanos, A. Ribeiro, G. J. Pappas, "Joint Control of Mobility and Communications in Networks of Robots," in Proc. 49th IEEE Conf. on Decision and Control, Dec. 2010
- [60] Van Tuan Le, Noury Bouraqadi, Serge Stinckwich, Victor Moraru, and Arnaud Doniocy Making Networked Robots Connectivity-Aware, ICRA'09 Proceedings of the 2009 IEEE international conference on Robotics and Automation. Pages 1835-1840, 2009.
- [61] <http://wapiti.telecom-lille1.eu/commun/ens/peda/options/st/rio/pub/exposes/exposesrio2008-ttffa2009/Leravallec-Schrifve/index.html>
- [62] A. SALMAN, Q. JUNAID, B. ADEEL, « Routing Protocols in Delay Tolerant Networks – A Survey », 6th International Conference on Emerging Technologies (ICET), October 2010, Islamabad, Pakistan. Pages 70-75, 2010.
- [63] H. GONG, L. YU, « Study on Routing Protocols for Delay Tolerant Mobile Networks », Hindawi Publishing Corporation International Journal of Distributed Sensor Networks Volume 2013, Article ID 145727, 16 pages

Références bibliographiques

- [64] Changqing Wang, Bang Wang and Wenyu Liu, A Survey: Movement Strategies for Improving Barrier Coverage in Wireless Sensor Networks, Communication Technology (ICCT), 2011 IEEE 13th International Conference, pages 938 – 943, 25-28 Sept. 2011.
- [65] S. JAIN, K. FALL, R. PATRA, « Routing in a Delay Tolerant Network », Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications, 2004
- [66] R-S. MANGRULKAR, M. ATIQUE, « Routing Protocol for Delay Tolerant Network: A Survey and Comparison », Communication Control and Computing Technologies (ICCCCT), 2010 IEEE International Conference, pages 210 – 215, 2010.
- [67] Y. ZHU, B. XU, X. SHI, Y. WANG, « A Survey of Social-Based Routing in Delay Tolerant Networks: Positive and Negative Social Effects », IEEE COMMUNICATIONS SURVEYS & TUTORIALS, VOL. 15, NO. 1, FIRST QUARTER 2013
- [68] A. ELWHISHI, P-H. HO, K. NAIK, B. SHIHADA, « A Novel Message Scheduling Framework for Delay Tolerant Networks Routing », IEEE transactions on parallel and distributed systems, VOL. 24, NO. 5, MAY 2013
- [69] A. LINDGREN, A. DORIA, O. SCHELEN, « Routing in Intermittently Connected Networks using a Probabilistic Approach », Service Assurance with Partial and Intermittent Resources, pages 239–254, 2004
- [70] C. MASCOLO, M. MUSOLESI, S. HAILES, « Adaptive Routing for Intermittently Connected Mobile Ad Hoc Networks », Sixth IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks. WoWMoM, 2005
- [71] J. Burgess, B. Gallagher, D. Jensen, B. Neil Levine, « MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks », INFOCOM 2006. 25th IEEE International Conference on Computer Communications, pages 1-11, 2006.
- [72] M. ATIQUE, R-S. MANGRULKAR, « Design and Development of Delay Tolerant Network : A Case Study », international conference on engineering, NUICONE-2012, 06-08 DECEMBER, 2012
- [73] L-F. XIE, P- H-J. CHONG, Y-L. GUAN, B-C. NG, « Routing for Group Mobility in Mobile Ad Hoc Networks with Delay-Tolerant Approach ». Wireless and Mobile Networking Conference (WMNC), 2010 Third Joint IFIP, pages 1-6, 13-15 Oct. 2010.
- [74] J. OTT, D. KUTSCHER, C. DWERTMANN, « Integrating DTN and MANET routing ». SIGCOMM workshop on Challenged networks, pages 221–228, 2006.
- [75] C. MASCOLO, M. MUSOLESI, « SCAR: Contextaware Adaptive Routing in Delay Tolerant Mobile Sensor Networks », international conference on Wireless communications and mobile computing, pages 533-538, 2006.
- [76] David E. Goldberg. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1989.
- [77] R. Kammarti, « Approches évolutionnistes pour la résolution du 1-pdptw statique et dynamique », Thèse présentée en vue d'obtenir le grade de docteur en spécialité : automatique et informatique industrielle, école centrale de Lille, université des sciences et technologies de Lille, 2006
- [78] S. Kirkpatrick , C. D. Gellatt Jr. and M. P. Vecchi, « Optimization by simulated annealing », 1982 : IBM Thomas J. Watson Research Center
- [79] F. Glover, “Tabu Search, Part I”, ORSA Journal on Computing 1(3), pp. 190-206. 1989
- [80] F. Glover, “Tabu Search, Part II”, ORSA Journal on Computing 2(1), pp. 4-32. 1990
- [81] K. YAGOUBI, « Evolution de design de circuit logique », mémoire de fin d'études en vue de l'obtention du titre d'ingénieur civil informaticien, Faculté des Sciences appliquées, Université libre de Bruxelles, 2008
- [82] Krishnanand, K.R, Nayak, S.K. ; Panigrahi, K.B. ; Rout, P.K , « Comparative study of five bio-inspired evolutionary optimization techniques », Nature & Biologically inspired computing, 2009. pages 1231-1236, IEEE. 2009
- [83] A. Clorni, M. Dorigo et V. Maniezzo, « Distributed optimization by ant colonies », in Proceedings of the first European conference on artificial life. Pages 134-142. 1991
- [84] C. Blum and D. Merkle, « Swarm Intelligence: Introduction and Applications », Springer-Verlag Berlin Heidelberg 2008
- [85] Mohammed El-Abd, « A Hybrid ABC-SPSO Algorithm for Continuous Function Optimization », IEEE 978-1-61284-052-9/11. 2011

Références bibliographiques

- [86] N. Kukururu, B-Rao Thella and R-L Davuluri, « Sensor Deployment Using Particle Swarm Optimization », *International Journal of Engineering Science and Technology* Vol. 2(10), 2010, 5395-5401
- [87] M. F. Tasgetiren, Y. Liang , M. Sevkli , G. Gencyilmaz , “A particle swarm optimization algorithm for makespan and total flowtime minimization in the permutation flowshop sequencing problem”. *European journal of operational research*, Vol. 177, pp.1930-1947. 2007
- [88] R. Brits, A.P. Engelbercht , F. Van der Bergh , “Location multiple optima using particle swarm optimization”, Elsevier, *Applied Mathematics and Computer*, 189, pp.1859-1883. 2007.
- [89] T. M. Blackwell, “Particle swarm optimization in dynamic environments”. *Evolutionary Computation in Dynamic and Uncertain Environments Studies in Computational Intelligence Volume 51*, 2007, pp 29-49
- [90] T.M. Blackwell , P.J Bentley “Dynamic search with charged swarm”, *Genetic and Evolutionary Computation, GECCO, Lecture Notes in Computer Science*, 2723, pp.1-12
- [91] T.M. Blackwell , J. Branke “Multi-swarms Exclusion, and Anti-Convergence in Dynamic Environments” , *Evolutionary Computation, IEEE Transactions on* (Volume:10 , Issue: 4) pages 459 - 472 , Aug. 2006.
- [92] J. W. Wang, X. W. Wang, M. Huang. An Intelligent QoS Multicast Routing Algorithm under Inaccurate Information. *Computational Intelligence and Security*. In 2006 International Conference on Publication, pages 073-1077, 2006
- [93] J. W. Wang, X. W. Wang, M. Huang. A Hybrid Intelligent QoS Multicast Routing Algorithm in NGI. In *Sixth International Conference on Parallel and Distributed Computing Applications and Technologies (PDCAT'05)*, pages 723-727, 2005
- [94] T. R. Gopalakrishnan Nair and K. Sooda, « Particle Swarm Optimization for Realizing Intelligent Routing in Networks with Quality Grading », *7th International Conference on Wireless Communication, Networking and Mobile Computing*, Wuhan, China, 2011
- [95] T. LI, Z. ZHAO, Z. GE, « An Adaptive Particle Swarm Optimization Algorithm for Anycast Routing », *Journal of Computational Information Systems* 7:5 (2011) 1559-1566, 2011
- [96] J. Zhuo, Yu-z Chen and Yi-p Chen, « a novel multi-objective optimization algorithm based on discrete pso for qos multicast routing in wireless mesh networks », *International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS 2010)* December 6-8, 2010
- [97] C. Blum and D. Merkle, « *Swarm Intelligence: Introduction and Applications* », Springer-Verlag Berlin Heidelberg, 2008
- [98] G. Onwubolu and M. Clerc. Optimal path for automated drilling operations by a new heuristic approach using particle swarm optimization. *International Journal of Production Research*, 42(3/01):473–491, February 2004
- [99] A. Salman, A. Imtiaz, and S. Al-Madani. Particle swarm optimization for task assignment problem. In *IASTED International Conference on Artificial Intelligence and Applications (AIA 2001)*, Marbella, Spain, 2001
- [100] J. Kennedy, R. C. Eberhart, and Y. Shi. *Swarm Intelligence*. Morgan Kaufmann Publishers, San Francisco, CA, 2004
- [101] M. Settles, B. Rodebaugh, and T. Soule. Comparison of genetic algorithm and particle swarm optimizer when evolving a recurrent neural network. In *Genetic and Evolutionary Computation Conference 2003 (GECCO 2003)*, pages 151– 152, Chicago, USA, 2003
- [102] E.S. Correa, A. Freitas, and C.G. Johnson. A new discrete particle swarm algorithm applied to attribute selection in a bioinformatics data set. In *GECCO 2006: Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation*, volume 1, pages 35–42, Seattle, Washington, USA, 2006. ACM Press
- [103] S. Katare, A. Kalos, and D. West. A hybrid swarm optimizer for efficient parameter estimation. In *Proceedings of the 2004 Congress on Evolutionary Computation CEC 2004*, pages 309–315. IEEE Press, 2004
- [104] G. K. Venayagamoorthy. Optimal control parameters for a UPFC in a multimachine using PSO. In *Proceedings of the 13th International Intelligent Systems Application to Power Systems 2005*, pages 488–493, 2005
- [105] C.-Y. Chen and F. Ye. Particle swarm optimization algorithm and its application to clustering analysis. In *IEEE International Conference on Networking, Sensing and Control*, volume 2, pages 789–794, 2004
- [106] A. Carlisle and G. Dozier. Adapting particle swarm optimization to dynamic environments. In *the Proceedings of the International Conference on Artificial Intelligence (ICAI 2000)*, pages 429–434, Las Vegas, Nevada, USA, 2000

Références bibliographiques

- [107] R. C. Eberhart and Y. Shi. Tracking and optimizing dynamic systems with particle swarms. In Proc. the 2001 Congress on Evolutionary Computation (CEC 2001), pages 94–100. IEEE Press, 2001
- [108] J. Moore and R. Chapman. Application of particle swarm to multiobjective optimization. Department of Computer Science and Software Engineering, Auburn University, 1999
- [109] J. E. Fieldsend and S. Singh. A multiobjective algorithm based upon particle swarm optimisation, an efficient data structure and turbulence. In Proceedings of the 2002 UK Workshop on Computational Intelligence, pages 37–44, Birmingham, UK, September 2002
- [110] X. Hu and R. Eberhart. Multiobjective optimization using dynamic neighborhood particle swarm optimization. In Proceedings of the 2002 Congress on Evolutionary Computation CEC 2002, pages 1677–1681. IEEE Press, 2002
- [111] K. Parsopoulos and M. Vrahatis. Particle swarm optimization method in multiobjective problems. In Proceedings of the 2002 ACM Symposium on Applied Computing (SAC 2002), pages 603–607. Madrid, Spain, ACM Press, 2002
- [112] A.Lemouari, « Systèmes Adaptatifs et Apprentissage Incrémental: de l'inspiration biologique à la résolution collective des problèmes », thèse en vue de l'obtention de grade de doctorat en Science de l'Université de Constantine Spécialité : Informatique, 2009
- [113] F. Sempé, "auto-organisation d'une collectivité de robots", thèse en vue de l'obtention du doctorat de l'université Pierre et Marie Curie, 2006
- [114] M. Schwager, J.McLurkin, D.Rus, " Distributed coverage control with sensory feedback for networked robots". Robotics: Science and Systems II, pages 49-56, 2006.
- [115] James McLurkin et Jennifer Smith, Distributed Algorithms for Dispersion in Indoor Environments Using a Swarm of Autonomous Mobile Robots, livre, pp 399-408, Springer Japan, 2007.
- [116] Hayes, A., Martinoli, A., and Goodman, R. Distributed odor source localization. *IEEE Sensors*, 2(3):260–271. 2002
- [117] A. Makhoul, "Réseaux de capteurs : localisation, couverture et fusion de données", thèse pour obtenir le grade de docteur de l'université de Franche-Comté, 2008
- [118] M-J ALDON, « Capteurs et méthodes pour la localisation des robots mobiles », Techniques de l'Ingénieur, traité Informatique industrielle S 7 852. 2001
- [119] Keith Y. K. Leung, Timothy D. Barfoot, Hugh H. T. Liuz, « Distributed and Decentralized Cooperative Simultaneous Localization and Mapping for Dynamic and Sparse Robot Networks», IEEE International Conference on Robotics and Automation Shanghai International Conference Center May 9-13, 2011, Shanghai, China
- [120] S. I. Roumeliotis and G. A. Bekey, « Distributed multirobot localization », Robotics and Automation, IEEE Transactions on (Volume:18, Issue: 5) 2002.
- [121] A. Howard, "Multi-robot simultaneous localization and mapping using particle filters," The International Journal of Robotics Research vol. 25 no. 12 1243-1256, December 2006
- [122] R. Kurazume, S. Hirose, "An experimental study of a cooperative positioning system," Autonomous Robots Volume 8, Issue 1, pp 43-52, 2000
- [123] Y. Dieudonne, O. Labbani-Igbida, F. Petit, "On the solvability of the localization problem in robot networks," Robotics and Automation, 2008. ICRA, IEEE. 2008
- [124] T. Langendoen, N. Reijers, « Distributed localization in wireless sensor networks: a quantitative comparison. », he International Journal of Computer and Telecommunications Networking - Special issue: Wireless sensor networks, Volume 43 Issue4, 15 November 2003 Pages 499-518
- [125] K. Leung, T.Barfoot, H. Liu, "Distributed and Decentralized Cooperative Simultaneous Localization and Mapping for Dynamic and Sparse Robot Networks", IEEE International Conference on Robotics and Automation, pages 3841 – 3847, 2011.
- [126] V. Tuan Le, "Coopération dans les systèmes multi-robots : Contribution au maintien de la connectivité et à l'allocation dynamique de rôles", thèse en vue de l'obtention du doctorat de l'université Caen, 2010
- [127] R. Meraihi, G Grand, "Contrôle de topologie orienté stratégie basesur des routeurs mobiles dédiés / Strategyorientedtopology control based on mobile dedicatedrouters", Journal Annales Des Télécommunications Volume 61, Issue 5-6 , pp 602-626, 2006
- [128] Ramanathan, Ram, Rosales-Hain, R. "Topology control of multihop wireless networks using transmit power adjustment", INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies.Proceedings. IEEE (Volume:2), 2000
- [129] Errol L. Lloyd , Rui Liu , Madhav V. Marathe , Ram Ramanathan , S. S. Ravi, Algorithmic aspects of topology control problems for ad hoc networks, Proceedings of the 3rd ACM

Références bibliographiques

- international symposium on Mobile ad hoc networking & computing, June 09-11, 2002, Lausanne, Switzerland.
- [130] Wattenhofer, R. ; Li Li ; Bahl, P. ; Yi-Min Wang, Distributed topology control for power efficient operation in multihop wireless ad hoc networks, INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE (Volume:3), 1388 - 1397 vol.3, 2001.
- [131] Y. O. Yazır, R. Farahbod, A. Guitouni, S. Ganti, Y. Coady. "Adaptive Routing in Mobile Ad Hoc Networks Based on Decision Aid Approach" in Proceedings of the 8th ACM International Symposium on Mobility Management and Wireless Access (MOBIWAC), 2010, pp. 1-10
- [132] ANTOINE BAUTIN « Stratégie d'exploration multi robot fondée sur le calcul de champs de potentiels », thèse pour l'obtention du doctorat de l'université de Lorraine (spécialité informatique), l'université de Lorraine, 2013
- [133] Mei, Y., Y.-H. Lu, C. Lee, and Y. Hu (2006, may). Energy-efficient mobile robot exploration. In Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on, pp. 505-511
- [134] Andrew Howard, Lynne E. Parker, Gaurav S. Sukhatme, Experiments with a Large Heterogeneous Mobile Robot Team: Exploration, Mapping, Deployment and Detection, , The International Journal of Robotics Research May 2006 vol. 25 no. 5-6431-447
- [135] U. Witkowski, M. El-Habbal, S. Herbrechtsmeier, A. Tanoto, J. Penders, L. Alboul, and V. Gazi. Ad-hoc network communication infrastructure for multi-robot systems in disaster scenarios. The International Workshop on Robotics for Risky Interventions and Surveillance of the Environment, 2008.
- [136] Milan Erdelj, Tahiry Razafindralambo, and David Simplot-Ryl, Covering Points of Interest with Mobile Sensors, IEEE Transactions On Parallel And Distributed Systems, Vol. 24, No. 1, January 2013
- [137] Y. Shang, W. Rumi, Y. Zhang and M. Fromherz, Localization from Connectivity in Sensor Networks. IEEE Transactions on Parallel and Distributed Systems, 15(11):961-974, 2004
- [138] Maxim A. Batalin and Gaurav S. Sukhatme, Coverage, Exploration and Deployment by a Mobile Robot and Communication Network, Telecommunication Systems Journal, Special Issue on Wireless Sensor Networks, 26(2-4), 2004
- [139] M. N. Rooker and A. Birk, "Multi-robot exploration under the constraints of wireless networking," Control Engineering Practice, vol. 15, no. 4, pp. 435-445, 2007.
- [140] <https://ccl.northwestern.edu/netlogo>

Résumé

D'une part, la mobilité des nœuds représente une caractéristique clé dans les réseaux mobiles ad hoc (MANETs), d'autre part le routage dans ces mêmes réseaux est basé sur l'existence d'une connectivité debout-en-bout. Une forte mobilité entraîne souvent des problèmes de déconnexion. Restreindre la mobilité des nœuds pour le maintien de la connectivité n'est pas une solution valable dans tous les scénarios. L'exploration est un exemple d'application qui oblige les nœuds (robots ou capteurs) à s'éloigner les uns des autres autant que possible afin d'explorer rapidement l'ensemble de la zone ciblée. Mais, d'autre part, chaque nœud doit conserver la connectivité avec les membres de son équipe afin de pouvoir communiquer. A cet effet, les protocoles proposés pour les réseaux tolérant au délai (Delay Tolerant Networking (DTN)) sont considérés comme les plus appropriés à utiliser dans les réseaux avec des déconnexions fréquentes. Cela nous a motivé à proposer un protocole adaptatif permettant de basculer entre des protocoles orientés MANET et d'autres orientés DTN, guidé par l'état de connectivité du réseau ciblé : a) un MANET si les robots ne constituent pas les nœuds du réseau, ou b) un RANET (Robot Adhoc Network ou RANET) sinon. Cela Pour atteindre une qualité d'exploration aussi élevée que possibles. Un phénomène de routage adaptatif émerge au macro-niveau du système RANET permettant d'observer comment un réseau de robots arrive à surmonter le problème de déconnexion pour réussir une tâche d'exploration.

Mots clés : Connectivité, Mobilité, Communication, MANET, DTN, Routage adaptatif, RANET.

Abstract

Mobility is a key characteristic of nodes in Mobile Ad hoc Network (MANET) but at the same time MANET is based on the existing of end-to-end connectivity. The high mobility often resulting of connectivity problems. Restrict node's mobility to maintaining connectivity is not available solution in all scenarios. Exploration is an application example, which needs nodes (robots or sensors) to move away from each other as far as possible to explore quickly the whole target area. But, on the other hand each node must keep connectivity with its team members in order to communicate. At the same time, Delay Tolerant Networking (DTN) protocols are considered as the most suitable to use in networks with frequent disconnections. This, motivate us to suggest an adaptive protocol idea, allowing to switch between the MANET and DTN protocols, according to the connectivity situation of a particular MANET (Robot Adhoc Network or RANET). Thus, to achieve an exploration task with best possible performances. An adaptive routing phenomenon emerges at the high level of the RANET leading to overcome disconnection problems in cooperative robot exploration task.

Index Terms—connectivity, mobility, communication, MANET, DTN, adaptive routing, RANET.

ملخص

تعد الحركة من الصفات الأساسية لعقد شبكة MANET. لكن في نفس الوقت هذا النوع من الشبكات يعتمد على وجود اتصال بين المرسل و المستقبل. الحركة الكثيرة تتسبب في مشكل انقطاع الاتصال. تقييد حركة العقد لضمان الاتصال ليسا حلا ناجحاً دائماً. عملية الاستكشاف من المهام التي تستوجب تباعد العقد (روبوتات أو أجهزة استشعار) لاستكشاف المنطقة المقصودة سريعاً. لكن في نفس الوقت، يستوجب على العقد البقاء قريبين من بعضهم للحفاظ على الاتصال بينهم. في حالة الشبكات ذات الانقطاع المتكرر، تعد بروتوكولات DTN الأنسب للتواصل. هذا السبب دفع بنا لاقتراح فكرة بروتوكول متكيف يسمح بالتبديل بين بروتوكولات شبكة MANET و بين بروتوكولات شبكة DTN وفقاً لحالة الاتصال لشبكة MANET خاصة (شبكة الروبوتات اللاسلكية: RANET). هذا كله لتحقيق مهمة الاستكشاف مع أفضل أداء ممكن. ظاهرة التوجيه التكيفي يظهر على مستوى عالٍ من RANET للتغلب على مشكلة انقطاع الاتصال في عملية استكشاف من طرف الروبوتات المتعاونة.

الكلمات الرئيسية: الربط، التنقل، الاتصال، DTN، MANET، التوجيه المتكيف، RANET.