

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE  
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE  
SCIENTIFIQUE  
UNIVERSITÉ ABDERRAHMANE MIRA DE BÉJAÏA



FACULTÉ DES SCIENCES EXACTES  
DÉPARTEMENT DE RECHERCHE OPERATIONELLES  
EN VUE DE L'OBTENTION DU DIPLÔME DE MASTER EN  
MATHÉMATIQUES APPLIQUÉES  
OPTION : OPTIMISATION ET FIABILITÉ DES RÉSEAUX DE  
COMMUNICATION

**Thème**

**LA RECHERCHE DES NŒUDS  
CENTRAUX DANS LES RÉSEAUX  
AD-HOC**

*Présenté par :*

AIT OUAKLI RIAD KHEREDDINE SMAIL

*Soutenu devant le jury composé de :*

<i>Président</i>	Dr N.KHIMOUM	M.C.A	U. A/MIRA BÉJAÏA
<i>Rapporteur</i>	Dr ELSAKAAN NADIM	Docteur	U. A/MIRA BÉJAÏA
<i>Examinatrice</i>	Dr N.BERNINE	M.C.B	U. A/MIRA BÉJAÏA
<i>Examinatrice</i>	Mme O.ALLEL	Doctorante	U. A/MIRA BÉJAÏA

2023 – 2024

## *Remerciements*

Initialement, On remercie Dieu de nous avoir donné le courage et la volonté qui nous ont permis d'aborder ce modeste travail.

On tiens à exprimer en tout premier lieu notre profonde gratitude à notre encadreur, Monsieur " ELSAKAAN Nadim ", de nous avoir proposé ce sujet, On le remercie vivement pour ces précieux conseils, pour sa disponibilité et surtout pour sa patience.

On tiens à remercier tous les membres du jury qui nous ont fait l'honneur de bien vouloir juger ce travail et de l'enrichir par leurs remarques et critiques. nos remerciements vont aussi à l'ensemble des enseignants et du personnel du département de recherche opérationnelle.

Enfin, On remercie du fond du cœur nos familles et amies pour leurs amour, leurs tendresse et leurs soutien inconditionnels.

*✧ Dédicaces ✧*

*Je dédie ce travail :*

*A mon père et ma mère qui m'ont soutenu et encouragé durant ces années d'études.  
qu'ils trouve ici le témoignage de ma profonde reconnaissance.*

*A mes soeurs Amel et Bouchra et mon frère Didine qui ont partagé avec moi tout les  
moments d'émotion lors de la réalisation de ce travail.ils m'ont challeurement  
supporté et encouragé tout au long de mon parcours.*

*A celle qui ma encouragé et soutenu pendant toutes ces années .*

*A mes amis précisément AIT OUKLI Riad et a l'équipe akham djedi qui m'ont  
toujours encouragé ,et a qui je souhaite plus de succès.*

*A mes collègues, et tous ceux qui m'ont aidé de près ou de loin*

*KHEREDDINE Smail*

✧ *Dédicaces* ✧

*Je dédie ce travail :*

*Ma cher mère, pour son amour inconditionnel, ses encouragements constants et ses innombrables sacrifices.*

*Mon père, pour son soutien indéfectible, son affection et la confiance qu'il m'a accordée.*

*Ma chère sœur Meriem et mes frères Hichem, ainsi que Rahim, pour leur affection et leur soutien constants.*

*Tous les membres de ma famille, pour leur soutien et leurs encouragements tout au long de ce parcours.*

*Mes amis et collègues d'études, pour leur camaraderie, leur soutien moral et leurs précieux conseils tout au long de cette aventure académique.*

*En particulier, mon frère et ami, Khereddine Smail, pour son aide précieuse pendant mes études.*

*AIT OUAKLI Riad*

# Table des matières

---

Liste des figures . . . . .	V
Liste des algorithmes . . . . .	VI
Liste des tables . . . . .	VII
<b>Introduction générale</b>	<b>1</b>
<b>1 Réseaux ad hoc et problématique des nœuds centraux</b>	<b>4</b>
1.1 Introduction . . . . .	4
1.2 Les réseaux sans fil . . . . .	4
1.2.1 Mode de connexion des réseaux sans fil . . . . .	5
1.2.2 Comparaison entre un réseau avec infrastructure et un réseau sans infrastructure . . . . .	7
1.3 Les Réseau Ad Hoc . . . . .	7
1.3.1 Définition d'un réseau Ad Hoc . . . . .	8
1.3.2 Historique . . . . .	8
1.3.3 Modélisation d'un réseau Ad hoc . . . . .	9
1.3.4 Caractéristiques des réseaux Ad hoc . . . . .	10
1.3.5 Mode de communication dans les réseaux Ad hoc . . . . .	12
1.3.6 Application des réseaux Ad hoc . . . . .	12
1.3.7 Défis des réseaux Ad Hoc . . . . .	14
1.4 Routage dans les réseaux Ad Hoc . . . . .	14
1.4.1 Définition du routage . . . . .	15
1.4.2 Les algorithmes de routage . . . . .	15
1.4.2.1 Algorithme à vecteur de distance . . . . .	16
1.4.2.2 Algorithme à état de liaison . . . . .	16
1.4.3 Protocole de routage dans les réseaux ad hoc . . . . .	16

1.4.4	Classification des protocoles de routage dans les réseaux ad hoc	17
1.4.4.1	Protocole de routage proactifs :	18
1.4.4.2	Protocole de routage réactif	18
1.4.4.3	Protocole de routage hybrid :	19
1.4.5	Problème de routage dans les réseaux ad hoc	19
1.5	Concepts de centralité des nœuds	21
1.5.1	Mesures de centralité des nœuds	22
1.5.2	Algorithmes dédiés à la détection des nœuds centraux	25
1.5.2.1	Approches par composantes bi-connexes	25
1.5.2.2	Approches spectrales (valeurs propres de la matrice Laplacienne)	25
1.6	Conclusion	26
1.7	Problématique et positionnement de la thèse	27
<b>2</b>	<b>Méthodes approchées : heuristiques et métaheuristiques</b>	<b>29</b>
2.1	Introduction	29
2.2	Classification des approches d'optimisation	29
2.2.1	Méthodes exactes	30
2.2.2	Méthodes approchées	30
2.3	Heuristique	30
2.4	Métaheuristique	31
2.4.1	Les métaheuristiques à base de solution unique	32
2.4.1.1	Le recuit simulé	32
2.4.2	Les métaheuristiques à base de population de solutions	36
2.4.2.1	Les algorithmes génétiques	37
2.4.2.2	Les algorithmes de colonies de fourmis (Ant Colony Optimization : ACO)	39
2.5	Conclusion	41
<b>3</b>	<b>Approches heuristique et métaheuristique pour la détection de nœuds centraux</b>	<b>42</b>
3.1	Introduction	42
3.2	Méthodologie Heuristique	42
3.3	Méthodologie métaheuristique basée sur le Recuit Simulé	44
3.3.1	Algorithme de Recuit Simulé	44

---

3.3.2	Formules Mathématiques . . . . .	46
3.4	Conclusion . . . . .	47
<b>4</b>	<b>Évaluation et discussion des résultats</b>	<b>48</b>
4.1	Introduction . . . . .	48
4.2	Résultats de la méthodologie Heuristique . . . . .	48
4.2.1	Analyse des résultats pour différentes tailles de réseau . . . . .	49
4.2.2	Analyse comparative des résultats . . . . .	55
4.2.3	Interprétation et implications . . . . .	56
4.2.4	Limites de l'approche heuristique . . . . .	56
4.3	Résultats de la méthodologie Métaheuristique basée sur le Recuit Simulé	57
4.3.1	Processus d'optimisation . . . . .	57
4.3.2	Analyse des résultats . . . . .	60
4.4	Analyse comparative des deux approches . . . . .	61
4.4.1	Comparaison avec des mesures de centralité établies dans la lit- térature . . . . .	63
4.5	Conclusion . . . . .	64
	<b>Conclusion générale</b>	<b>65</b>
	<b>Bibliographie</b>	<b>70</b>
	<b>Résumé</b>	<b>71</b>
	<b>Abstract</b>	<b>72</b>

# Table des figures

---

1.1 Réseaux sans fil avec infrastructure . . . . .	5
1.2 Réseaux sans fil sans infrastructure (Ad hoc) . . . . .	6
1.3 Réseaux sans fil sans infrastructure(Ad hoc) . . . . .	8
1.4 Modélisation d'un réseau Ad hoc . . . . .	10
1.5 Mobilité des noeuds d'un réseau Ad hoc . . . . .	11
1.6 Domaine d'utilisation des réseaux ad hoc . . . . .	13
1.7 Classification des protocoles de routage pour réseaux Ad-hoc[19] . . .	17
1.8 Représentation d'un réseau ad hoc : à gauche, le réseau ad hoc avec toute les connexion ; à droite, le réseau après la suppression de 20% des noeuds. . . . .	21
1.9 Centralité d'intermédiarité des noeuds (La teinte, allant de rouge = 0 à bleu = max, indique l'intermédiarité des noeuds) . . . . .	24
2.1 Classification des métaheuristiques. . . . .	32
2.2 Comportement de la recherche locale (descente la plus raide) dans un paysage donné.)[10] . . . . .	34
2.3 Démarche d'un algorithme génétique. [6] . . . . .	38
4.1 Visualisation du réseau ad hoc de 50 noeuds. . . . .	49
4.2 Visualisation du réseau ad hoc de 50 noeuds. Les noeuds en rouge repré- sentent les noeuds centraux identifiés par l'algorithme heuristique. . . .	49
4.3 Visualisation du réseau ad hoc de 150 noeuds. . . . .	51
4.4 Visualisation du réseau ad hoc de 150 noeuds. Les noeuds en rouge re- présentent les noeuds centraux identifiés par l'algorithme heuristique . .	51
4.5 Visualisation du réseau ad hoc de 300 noeuds. . . . .	53

---

4.6	Visualisation du réseau ad hoc de 300 nœuds. Les nœuds en rouge représentent les nœuds centraux identifiés par l'algorithme heuristique . . .	53
4.7	Graphe initial montrant tous les nœuds et arêtes présents. . . . .	58
4.8	Graphe après suppression des nœuds, montrant les composantes connexes distinctes. . . . .	59
4.9	Graphe final, avec les nœuds supprimés en rouge, illustrant les composantes connexes. . . . .	59
4.10	Visualisation des nœuds centraux identifiés par les approches heuristique et métaheuristique . . . . .	62
4.11	Distribution des degrés des nœuds . . . . .	63

# Liste des algorithmes

---

1	Recuit simulé[30] . . . . .	33
2	Algorithme de recherche locale[10] . . . . .	35
3	Algorithme Génétique[24] . . . . .	39
4	Algorithme de Colonies de Fourmis[4] . . . . .	40
5	Identification des Nœuds Centraux dans un Réseau Ad Hoc . . . . .	43
6	Optimisation de la Centralité des Nœuds par Recuit Simulé . . . . .	46

# Liste des tableaux

---

1.1	Comparaison entre un réseau avec infrastructure et un réseau sans infrastructure . . . . .	7
4.1	Scores de centralité de 50 nœuds . . . . .	50
4.2	Scores de centralité de 150 nœuds . . . . .	52
4.3	Scores de centralité de 300 nœuds . . . . .	54

# Introduction générale

---

Les réseaux ad hoc constituent un champ d'étude fascinant et en constante évolution dans le domaine des télécommunications modernes. Ces systèmes de communication, caractérisés par leur nature décentralisée et leur capacité d'auto-organisation, sont composés de nœuds mobiles interconnectés sans recourir à une infrastructure fixe préétablie. Cette flexibilité intrinsèque des réseaux ad hoc ouvre la voie à une multitude d'applications dans des contextes variés[7].

En effet, ces réseaux trouvent leur utilité dans des scénarios aussi divers que la gestion des situations d'urgence, où la rapidité de déploiement est importante, les opérations militaires nécessitant une communication robuste et adaptable, ou encore les environnements industriels complexes requérant une connectivité flexible. Leur capacité à s'adapter rapidement aux changements topologiques et à fonctionner de manière autonome les rend particulièrement précieux dans des situations où les infrastructures de communication traditionnelles sont inexistantes, endommagées ou inadaptées[34].

Cependant, la nature même de ces réseaux, caractérisée par l'absence d'une infrastructure centrale, soulève des défis uniques et complexes. La gestion efficace des ressources limitées, l'optimisation des protocoles de routage et le maintien des performances globales du réseau dans un environnement en constante évolution constituent autant d'enjeux majeurs pour les chercheurs et les ingénieurs du domaine. Ces défis sont d'autant plus cruciaux que la qualité et la fiabilité des communications au sein des réseaux ad hoc peuvent avoir des implications directes sur la sécurité, l'efficacité opérationnelle, voire la vie des utilisateurs dans certains contextes critiques[27].

Au cœur de l'optimisation des réseaux ad hoc se trouve la problématique cruciale de l'identification des nœuds centraux. Ces nœuds, véritables piliers du réseau, jouent un rôle déterminant dans la connectivité globale et l'efficacité de l'acheminement des données. Leur identification précise représente un défi majeur, aux implications considé-

rables pour la performance et la résilience du réseau[32]. En effet, la nature dynamique et décentralisée des réseaux ad hoc rend particulièrement complexe la tâche d'évaluer l'importance relative de chaque nœud. La topologie du réseau évolue constamment, les nœuds entrent et sortent de la zone de couverture, et les liens entre eux se font et se défont au gré des mouvements et des interférences. Dans ce contexte mouvant, comment déterminer avec précision quels nœuds sont les plus critiques pour le maintien de la connectivité et l'efficacité du routage?

Cette question soulève plusieurs défis interconnectés :

1. La définition même de la centralité dans un réseau ad hoc : quels critères permettent de qualifier un nœud comme central ?
2. La conception d'algorithmes capables d'évaluer cette centralité de manière efficace et en temps réel, malgré les contraintes de ressources inhérentes aux réseaux ad hoc.
3. L'adaptation de ces méthodes à la nature dynamique du réseau, nécessitant une réévaluation constante de la centralité des nœuds.
4. La prise en compte de l'impact de la perte potentielle de nœuds centraux sur la robustesse globale du réseau.

Nos contributions à l'étude des réseaux ad hoc se concentrent particulièrement sur l'identification des nœuds centraux. Notre approche se distingue par l'utilisation de méthodes efficaces et une analyse approfondie adaptées à la modélisation complexe

Nos principales contributions sont les suivantes :

1. Proposition d'une nouvelle heuristique basée sur la profondeur des arborescences enracinées : Nous avons conçu et implémenté en Python une méthode originale pour évaluer la centralité des nœuds. Cette approche exploite la structure topologique du réseau pour identifier les nœuds stratégiquement positionnés.
2. Adaptation d'une métaheuristique : En parallèle, nous avons élaboré une métaheuristique qui évalue l'importance des nœuds en analysant l'impact de leur suppression sur la connectivité globale du réseau. L'implémentation en Python de cette approche a facilité l'intégration de techniques d'optimisation complexes et l'analyse de scénarios multiples de suppression de nœuds.
3. Analyse comparative des deux approches : Nous avons créé un environnement de simulation permettant une comparaison directe des deux méthodes. Cette modélisation a mis en lumière les forces et les limites de chaque approche.

Notre manuscrit s'articule autour de quatre chapitres principaux, chacun abordant un aspect essentiel de notre recherche sur l'identification des nœuds centraux.

Le premier chapitre, intitulé "Réseaux ad hoc et problématique des nœuds centraux", pose les fondements de notre étude. Il offre une vue d'ensemble des réseaux ad hoc, en couvrant leur historique, leurs caractéristiques, et les défis qu'ils présentent. Une attention particulière est portée au routage dans ces réseaux. Ce chapitre introduit ensuite le concept de centralité des nœuds, présentant les différentes mesures existantes et leur importance dans l'optimisation des réseaux ad hoc. Il se conclut par l'exposition détaillée de notre problématique de recherche, situant notre travail dans le contexte scientifique actuel.

Le deuxième chapitre, "Méthodes approchées : heuristiques et métaheuristiques", présente un panorama théorique des techniques d'optimisation. Nous y explorons les principes fondamentaux des heuristiques et des métaheuristiques d'un point de vue général. Ce chapitre aborde les concepts clés, les différentes catégories de méthodes, et leurs caractéristiques principales. Cette vue d'ensemble théorique fournit les bases nécessaires pour comprendre le contexte et les fondements des approches spécifiques que nous développons dans les chapitres suivants.

Le troisième chapitre, "Approches heuristique et métaheuristique pour la détection de nœuds centraux", constitue le cœur de notre contribution. Nous y présentons en détail nos deux approches : une heuristique basée sur la profondeur des arborescences enracinées et une métaheuristique évaluant l'importance des nœuds par l'impact de leur suppression. Ce chapitre met en lumière notre méthodologie, expliquant comment nous avons conçu et implémenté ces algorithmes en utilisant Python. Nous détaillons les concepts mathématiques sous-jacents, les structures de données utilisées, et les optimisations apportées pour assurer l'efficacité de nos méthodes dans le contexte des réseaux ad hoc.

Le quatrième et dernier chapitre, "Évaluation et Discussion des Résultats", est consacré à l'analyse critique de nos travaux. Nous y présentons les résultats obtenus par nos algorithmes heuristiques et métaheuristiques, offrant une évaluation comparative approfondie de leur efficacité et de leur pertinence dans divers scénarios de réseaux ad hoc. Ce chapitre inclut une discussion détaillée sur les performances de nos méthodes, leurs limites, et les perspectives qu'elles ouvrent pour de futures recherches. Nous mettons également en évidence les implications pratiques de nos résultats pour l'optimisation des réseaux ad hoc dans des applications réelles.

# Réseaux ad hoc et problématique des nœuds centraux

---

## 1.1 Introduction

Les réseaux ad hoc sont des systèmes de communication décentralisés formés par des nœuds mobiles interconnectés. Ils sont particulièrement adaptés aux situations où l'infrastructure de communication est absente ou impraticable.

Ces réseaux trouvent des applications dans divers domaines, tels que les situations d'urgence et les environnements urbains intelligents. Cependant, leur nature dynamique pose des défis uniques, notamment en ce qui concerne la gestion des nœuds centraux, essentiels à la performance du réseau.

Ce chapitre présente les caractéristiques des réseaux ad hoc et examine la problématique des nœuds centraux, ainsi que les méthodes pour les identifier et les gérer

## 1.2 Les réseaux sans fil

le réseau sans fil désigne l'utilisation des ondes hertziennes (signaux infrarouges ou radiofréquences) pour échanger des informations et des ressources entre des appareils. Divers types d'appareils sans fil sont disponibles aujourd'hui, tels que les terminaux mobiles, les PC de poche, les ordinateurs portables, les téléphones cellulaires, les assistants personnels numériques (PDAs), les capteurs sans fil et les récepteurs satellites, entre autres [2].

### 1.2.1 Mode de connexion des réseaux sans fil

Généralement, on distingue deux types d'architectures dans les réseaux sans fil : le mode sans infrastructure et le mode avec infrastructure sont les plus courants.

#### Mode avec infrastructure

Un réseau sans fil avec infrastructure est une architecture où tous les périphériques clients sans fil se connectent à un nœud central appelé point d'accès.

Le point d'accès fait office de concentrateur et de passerelle. En d'autres termes, plutôt que de communiquer directement entre eux, les clients échangent toutes leurs données via le point d'accès. Celui-ci relaie les transmissions entre les différents clients du réseau sans fil. Il assure également la connexion avec d'autres réseaux tels qu'Internet ou un réseau local câblé.[28]



FIGURE 1.1 – Réseaux sans fil avec infrastructure

#### Mode sans infrastructure (Ad hoc)

Un réseau sans infrastructure ad hoc, est un ensemble de nœuds mobiles autonomes qui coopèrent entre eux pour communiquer sans nécessiter une infrastructure préexistante ou une gestion centralisée. Le réseau se forme spontanément lorsque les nœuds sont à portée les uns des autres. Deux modes de communication sont possibles : la transmission ad hoc lorsque les nœuds sont à portée directe, et la transmission multi-

sauts lorsque des relais sont nécessaires. L'absence d'infrastructure filaire oblige les unités mobiles à agir comme des routeurs, facilitant ainsi la découverte et la maintenance des chemins de communication pour les autres nœuds du réseau[18]

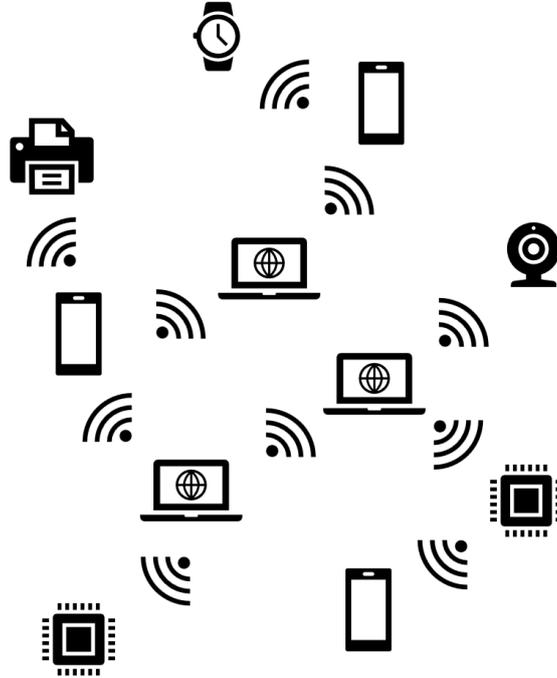


FIGURE 1.2 – Réseaux sans fil sans infrastructure (Ad hoc)

### 1.2.2 Comparaison entre un réseau avec infrastructure et un réseau sans infrastructure

Les réseaux sans fil peuvent fonctionner de deux façons différentes : avec infrastructure ou ad hoc. Le tableau 1.1 récapitule les différences entre les réseaux sans fil avec infrastructure et les réseaux sans fil ad hoc.

Critère	Réseau avec infrastructure	Réseau sans infrastructure (ad hoc)
Architecture	Utilise un point d'accès central	Pas de point d'accès central
Connectivité	Tous les nœuds communiquent via le point d'accès	Les nœuds communiquent directement entre eux
Exemples	Réseaux Wi-Fi domestiques/entreprise	Réseaux maillés sans fil, réseaux de capteurs
Gestion	Centralisée via le point d'accès	Décentralisée, gérée par chaque nœud
Contrôle	Contrôle centralisé du réseau	Pas de contrôle centralisé
Flexibilité	Moins flexible, dépend du point d'accès	Plus flexible, configuration ad hoc
Point de défaillance	Point d'accès = point de défaillance unique	Pas de point de défaillance unique
Configuration	Configuration plus complexe	Configuration plus simple
Sécurité	Sécurité gérée au niveau du point d'accès	Sécurité gérée au niveau de chaque nœud
Portée	Limitée par la portée du point d'accès	Potentiellement plus grande portée en multisaut

TABLE 1.1 – Comparaison entre un réseau avec infrastructure et un réseau sans infrastructure

## 1.3 Les Réseau Ad Hoc

Grâce aux avancées récentes dans les technologies de communication sans fil et à l'émergence de dispositifs informatiques portables comme les ordinateurs portables, les chercheurs s'orientent désormais vers l'objectif ultime des réseaux : offrir un accès à l'information en tout lieu et à tout moment.

Les réseaux ad hoc ont pour but d'élargir les concepts de mobilité à l'ensemble des éléments composant l'environnement réseau.

### 1.3.1 Définition d'un réseau Ad Hoc

Un réseau sans fil Ad hoc est un ensemble de deux appareils ou plus dotés de capacités de communication sans fil, capables de communiquer entre eux directement ou via des nœuds intermédiaires. Ces réseaux sont auto-organisés et adaptatifs, ne nécessitant aucune intervention administrative. Les appareils ad hoc, pouvant prendre diverses formes, doivent être en mesure de détecter les autres appareils pour faciliter la communication. Étant dépourvus d'infrastructure fixe, ces réseaux n'exigent ni stations de base radio ni routeurs fixes. Toutefois, la mobilité implique des ajustements dans les données de routage. La variété des appareils ad hoc entraîne des variations dans la capacité de la batterie, la consommation d'énergie étant un défi majeur. [7]



FIGURE 1.3 – Réseaux sans fil sans infrastructure(Ad hoc)

### 1.3.2 Historique

L'histoire des réseaux ad hoc remonte aux débuts des années 1970, trouvant ses premières applications dans le domaine militaire pour améliorer et garantir les communications sans fil. À cette époque, la Defense Advanced Research Project Agency (DARPA) des États-Unis a été à l'avant-garde de la création de ces réseaux mobiles sans infrastructure fixe.

Les premiers prototype étaient conçus pour fonctionner dans des conditions difficiles, permettant une communication efficace entre des dispositifs mobiles sans dépendre d'une infrastructure fixe,

Au milieu des années 1980, des progrès significatifs ont été réalisés avec le déve-

loppement du concept de "Survivable Radio Networks" (SURAN) par la DARPA pour surmonter les lacunes des premiers projets projet et permettre au réseau de supporter plusieurs milliers de nœuds mobiles, gérés par des protocoles dotés de mécanismes radio simples.

À partir des années 1990, le groupe de travail Mobile Ad Hoc Network (MANET) a intensifié ses efforts de recherche dans ce domaine. Le développement des technologies radio, en particulier la norme IEEE 802.11 et ses dérivées, a joué un rôle important dans l'évolution des réseaux ad hoc en permettant une connectivité sans fil plus efficace entre les dispositifs. Depuis lors, les réseaux ad hoc ont continué à évoluer pour répondre aux besoins croissants de connectivité flexible et décentralisée dans divers domaines, allant au-delà du militaire pour inclure des applications civiles, industrielles et de recherche.[34]

### 1.3.3 Modélisation d'un réseau Ad hoc

Un réseau ad hoc peut être modélisé par un graphe  $G_t = (V_t, E_t)$ , où  $V_t$  représente l'ensemble des nœuds (par exemple, les unités ou les hôtes mobiles) du réseau et  $E_t$  modélise l'ensemble des connections qui existent entre ces nœuds. Si  $e = (u, v) \in E_t$ , cela signifie que les nœuds  $u$  et  $v$  sont en mesure de communiquer directement à l'instant  $t$ . La topologie du réseau peut changer à tout moment, elle est donc dynamique et imprévisible, ce qui fait que la déconnexion des unités est très fréquente. La figure représente un réseau ad hoc de 8 unités mobiles sous forme d'un graphe.[19]

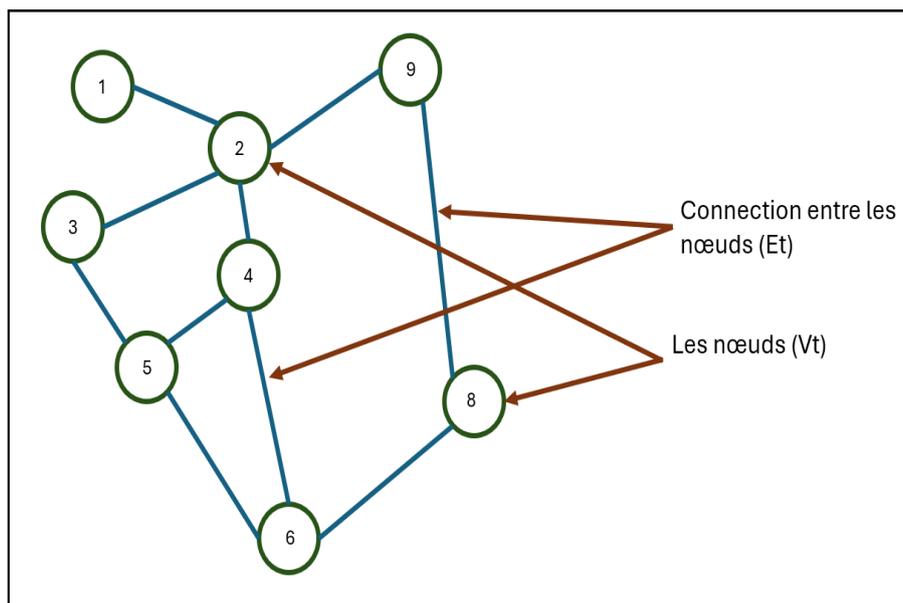


FIGURE 1.4 – Modélisation d'un réseau Ad hoc

### 1.3.4 Caractéristiques des réseaux Ad hoc

Les réseaux Ad hoc possèdent non seulement les mêmes caractéristiques que les réseaux sans fil, mais également un certain nombre de caractéristiques qui leur sont propres et les différencient des autres. Nous pouvons citer quelques fonctionnalités clés :[19]

1. **Absence d'infrastructure** : Les réseaux Ad hoc fonctionnent sans nécessiter de station de base ou de point d'accès centralisé. Chaque appareil opère de manière autonome, agissant à la fois en tant qu'émetteur et récepteur dans le réseau, sans nécessiter de point d'attache fixe à l'ensemble du réseau.
2. **Topologie réseau dynamique** : La mobilité des appareils dans les réseaux mobiles ad hoc conduit à une topologie de réseau dynamique. Les changements dans la disposition spatiale des appareils peuvent survenir rapidement et de manière imprévisible, ce qui entraîne des modifications constantes dans les connexions entre les nœuds du réseau.

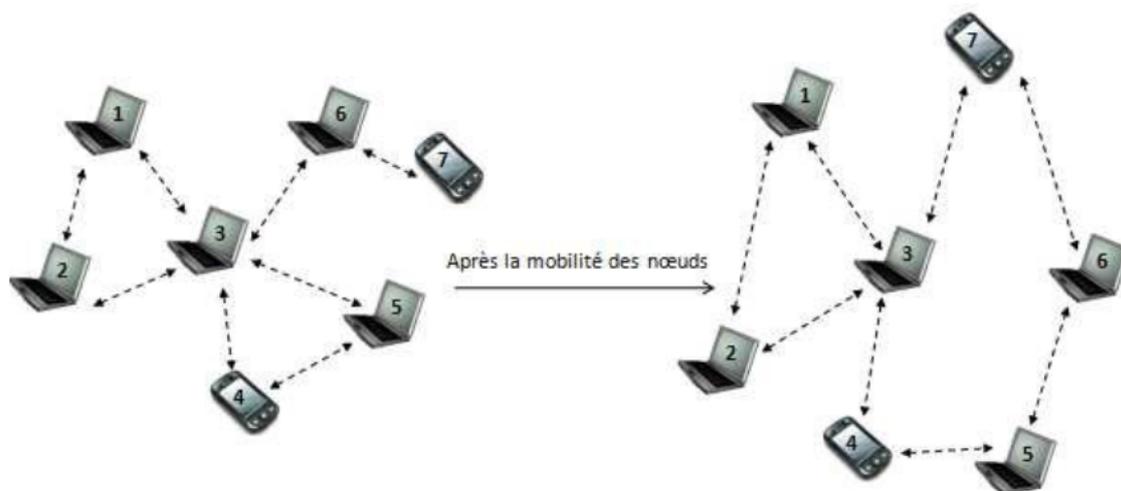


FIGURE 1.5 – Mobilité des noeuds d'un réseau Ad hoc

3. **Ressources limitées** : Les appareils mobiles dans les réseaux ad hoc dépendent de sources d'énergie limitées, telles que les batteries, pour leur fonctionnement. La durée de vie de ces batteries est limitée, et leur épuisement est accéléré par des opérations telles que la transmission de données, la réception et les calculs complexes. La gestion efficace de l'énergie est donc essentielle pour prévenir les pannes et prolonger la durée de vie des appareils.
4. **Réseau à faible coût** : Les réseaux câblés sont coûteux car ils nécessitent un câblage et le déploiement d'une infrastructure. En revanche, les réseaux Ad hoc peuvent être déployés n'importe où, en particulier dans les zones où les réseaux câblés ne sont pas praticables pour des raisons géographiques. Ainsi, les réseaux ad hoc offrent une alternative économique.
5. **Environnements hostiles** : Les réseaux ad hoc sont particulièrement pratiques dans des environnements difficiles d'accès, comme les régions montagneuses ou les zones de combat.
6. **Infrastructure sans fil** : Dans des situations locales telles que des conférences ou des réunions, où un réseau câblé n'est pas nécessaire, les réseaux ad hoc peuvent être utilisés. Ils offrent une solution pratique pour les communications internes sans avoir besoin de recourir à un réseau câblé.

### 1.3.5 Mode de communication dans les réseaux Ad hoc

Dans les réseaux ad hoc, les échanges de données se déroulent selon trois principaux modes de communication :[23]

- La communication **”point à point”** ou unicast consiste à spécifier le nœud suivant sur la route d’un paquet de données, ce qui permet d’acheminer efficacement les informations vers une destination spécifique.
- La communication **”multipoint”** ou multicast permet à chaque nœud source de transmettre ses paquets à un groupe spécifique de nœuds, facilitant ainsi la diffusion d’informations à plusieurs destinataires simultanément.
- La communication par diffusion ou **”broadcast”** implique qu’un nœud source envoie ses paquets à tous les nœuds voisins. Chaque nœud mobile réémet alors les paquets qu’il reçoit à ses voisins, assurant ainsi une propagation étendue des données dans le réseau jusqu’à ce qu’elles atteignent leur destination. Ce mode de communication est particulièrement efficace pour diffuser des informations à l’ensemble du réseau, mais peut également entraîner une utilisation intensive des ressources en cas de trafic important.

### 1.3.6 Application des réseaux Ad hoc

Les réseaux ad hoc sont employés dans diverses applications où l’établissement d’une architecture centralisée est difficile voire impossible. Leur robustesse, leur faible coût et leur déploiement rapide les rendent adaptés à un large éventail d’utilisations, notamment :[34]

- **Utilisations militaires :** Les réseaux ad hoc sont spécifiquement conçus pour répondre aux besoins opérationnels et militaires, étant particulièrement adaptés aux environnements hostiles en raison de leur capacité à être rapidement déployés et à s’adapter dynamiquement. Ces réseaux impliquent souvent des soldats et des véhicules blindés comme nœuds de communication.

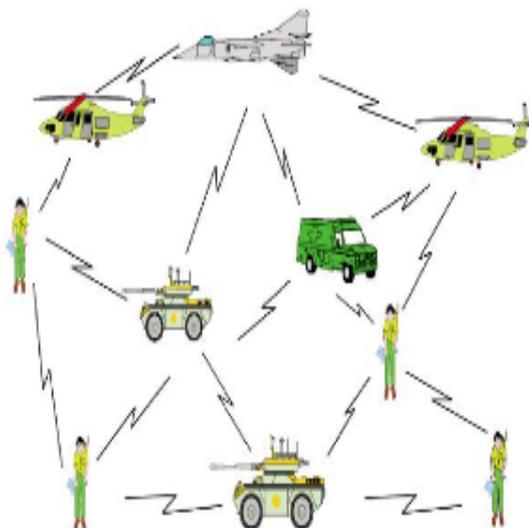


FIGURE 1.6 – Domaine d'utilisation des réseaux ad hoc

La figure 1.4 représente l'utilisation militaire des réseaux ad hoc, mettant en évidence leur capacité à établir des communications en environnements tactiques sans infrastructure préexistante.

- **Opérations de secours :** Les réseaux sans fil ad hoc sont également employés dans les opérations de secours, notamment lors de catastrophes naturelles telles que les tremblements de terre. Leur déploiement rapide sur le terrain facilite la communication entre les équipes de secours, contribuant ainsi à l'efficacité des opérations de sauvetage.
- **Réseaux de capteurs sans fil (WSN) :** Les réseaux de capteurs sans fil (WSNs) sont parmi les applications les plus marquantes du paradigme de communication ad hoc, ces réseaux sont composés de nombreux capteurs autonomes, chaque nœud capteur est fait d'une unité d'acquisition, d'un processeur, d'une interface de communication radio et d'une batterie. A l'instar des stations de collecte, les nœuds capteurs ont une puissance de calcul, une capacité de stockage et une autonomie de batterie très limitées. Les nœuds dans un WSN sont typiquement statiques et d'une forte densité. L'interconnexion sans fil entre les nœuds capteur s'effectue généralement via les standards Bluetooth ou Zigbee, dispersés dans une zone géographique donnée, capables de détecter diverses conditions environnementales telles que la température, la pression, l'humidité ou les vibrations. Grâce à leurs capacités de communication sans fil, les données collectées par ces capteurs peuvent être acheminées vers un nœud central pour

y être analysées et traitées.

- **Utilisation en entreprise :** La facilité de déploiement et le coût réduit des réseaux ad hoc suscitent un intérêt croissant dans le monde des affaires. Ces réseaux permettent une grande mobilité des employés, favorisent le partage des données et facilitent l'organisation de réunions et de conférences interactives. Par exemple, lors d'une réunion, un présentateur peut communiquer avec tous les participants et encourager un débat interactif.

### 1.3.7 Défis des réseaux Ad Hoc

Les réseaux ad hoc héritent des défis classiques des réseaux sans fil, tout en confrontant également de nouveaux problèmes et défis uniques. Parmi les principaux enjeux de ces réseaux, on peut citer :[27]

- La complexité de l'analyse des réseaux ad hoc réside dans leur topologie imprévisible, due aux déplacements fréquents des nœuds, entraînant des changements rapides dans la configuration du réseau.
- Les capacités limitées des nœuds posent un défi majeur, car ajuster la portée de communication pour garantir la connectivité tout en minimisant la consommation d'énergie nécessite un équilibre délicat.
- Le taux élevé d'erreurs de transmission est un problème important, car le risque de collisions augmente avec le nombre croissant de nœuds partageant le même canal de communication, surtout lorsque la portée des communications est étendue.
- La sécurité constitue également un défi de taille, car les réseaux ad hoc sont vulnérables aux accès non autorisés en raison de leur nature sans fil et de l'absence d'une infrastructure dédiée à la sécurité. Cela soulève des préoccupations quant à la confidentialité des données échangées entre les nœuds, ce qui suscite l'intérêt des chercheurs et des experts en sécurité des réseaux ad hoc.

## 1.4 Routage dans les réseaux Ad Hoc

Le routage représente un aspect fondamental et critique dans les réseaux ad hoc. Contrairement aux réseaux câblés ou aux réseaux sans fil traditionnels, les nœuds mobiles au sein des réseaux Ad hoc doivent assurer eux-mêmes les fonctions de routage

pour acheminer les données à travers le réseau de manière efficace. La nature dynamique de la topologie des réseaux ad hoc, due à la mobilité des nœuds, ainsi que les contraintes en termes de ressources énergétiques et de capacité de traitement, posent des défis uniques pour la conception de protocoles de routage performants.

### 1.4.1 Définition du routage

Le routage est le processus de transfert des paquets de données d'un expéditeur vers le ou les destinataires appropriés via un réseau de télécommunications. C'est une fonction essentielle qui permet l'acheminement efficace des données à travers des réseaux informatiques interconnectés.[13]

### 1.4.2 Les algorithmes de routage

Un algorithme de routage est une méthode systématique qui guide la sélection des chemins à travers un réseau pour le transfert de paquets de données. Son objectif principal est d'identifier les routes optimales en fonction de critères prédéfinis tels que le nombre de sauts, la bande passante, le délai, le coût, ou d'autres métriques pertinentes.

Pour ce faire, les algorithmes de routage recueillent des informations sur l'état du réseau, comme sa topologie et les caractéristiques des liens, et les utilisent pour établir et mettre à jour des tables de routage. Ces tables contiennent les données nécessaires pour associer les destinations aux interfaces de sortie appropriées, facilitant ainsi le transfert efficace des paquets de données par les routeurs.[26]

Il existe plusieurs types d'algorithmes de routage, chacun ayant ses propres caractéristiques, avantages et inconvénients. Voici quelques-unes des principales catégories :

#### 1.4.2.1 Algorithme à vecteur de distance

Un algorithme de routage à vecteur de distance est un algorithme itératif où chaque nœud construit une table de vecteurs contenant les distances les plus courtes vers d'autres nœuds ainsi que l'estimation du chemin le plus court vers ces nœuds. Les nœuds échangent régulièrement leurs tables de vecteurs avec leurs voisins.[13]

#### 1.4.2.2 Algorithme à état de liaison

Un algorithme de routage à état de liaison permet à chaque routeur de créer une carte du réseau en collectant des informations de connexion de tous les autres routeurs. Ensuite, chaque routeur utilise cette carte et un algorithme comme celui de Dijkstra pour trouver les chemins les plus courts vers chaque destination.[11]

### 1.4.3 Protocole de routage dans les réseaux ad hoc

Les protocoles de routage utilisés dans les réseaux ad hoc sont basés sur des principes fondamentaux tels que l'inondation, le vecteur de distance, le routage à la source et l'état de lien. À partir de la normalisation des réseaux Ad hoc, deux grandes catégories de protocoles de routage ont émergé : les protocoles proactifs, qui établissent les routes à l'avance, et les protocoles réactifs, qui recherchent les routes à la demande. D'autres classes de protocoles incluent les protocoles hybrides, qui combinent des aspects réactifs et proactifs, ainsi que les protocoles géographiques, hiérarchiques, de qualité de service et de multicast.[11]

## 1.4.4 Classification des protocoles de routage dans les réseaux ad hoc

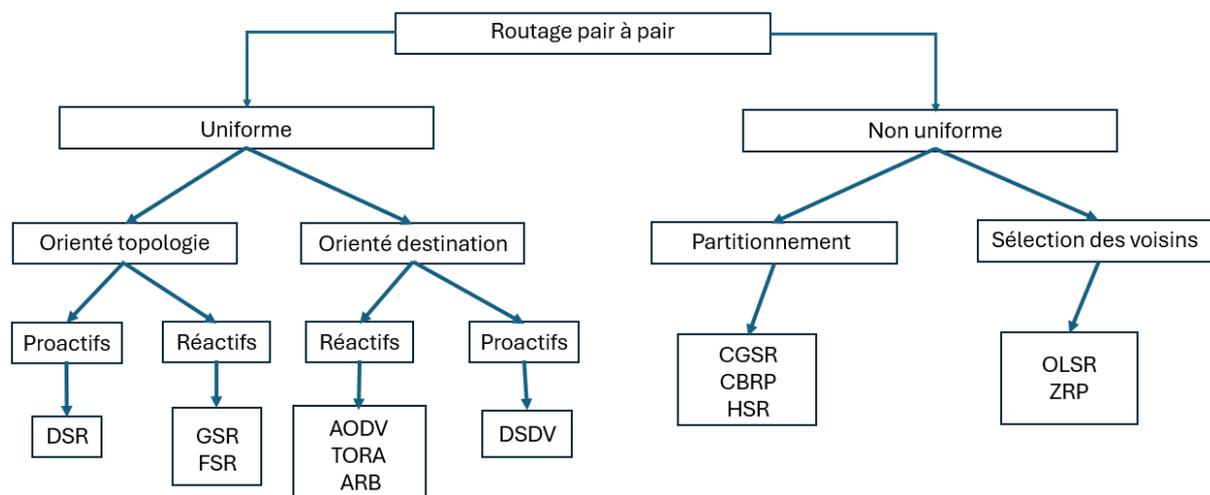


FIGURE 1.7 – Classification des protocoles de routage pour réseaux Ad-hoc[19]

La figure 1.5 propose une classification des protocoles de routage utilisés dans les réseaux ad hoc. Ces protocoles se distinguent principalement par le degré d’engagement des nœuds dans le processus de routage.

- **Uniforme** : tous les nœuds du réseau jouent le même rôle pour la fonction de routage
- **Non uniforme** : Le réseau est organisé selon une structure hiérarchique où seuls certains nœuds sont responsables du routage.
- **Protocoles à sélection de voisins** Chaque nœud délègue la fonction de routage à un groupe restreint de ses voisins directs.
- **Protocoles à partitionnement** Le réseau est divisé en zones où le routage est géré par un seul nœud maître.
- **Protocoles orientés topologie** Chaque nœud utilise les informations sur l’état de ses connexions avec ses voisins, qu’il transmet ensuite aux autres nœuds pour améliorer leur compréhension de la topologie du réseau.
- **Protocoles orientés destination** ces protocoles maintiennent des informations pour chaque nœud de destination, notamment le nombre de nœuds qui les séparent (la distance) et éventuellement la première direction à prendre pour les

atteindre.

#### 1.4.4.1 Protocole de routage proactifs :

Dans cette catégorie, aussi appelée diffusion de table, chaque nœud du réseau maintient une table de routage. Les messages de mise à jour sont envoyés à tous les nœuds chaque fois que le réseau change. Les protocoles de cette catégorie incluent DSDV, WRP, OLSR, GSR, FSR, HSR, CGSR et LSR.

Les protocoles de routage proactifs visent à maintenir les meilleurs chemins vers toutes les destinations possibles. Pour cela, ils échangent régulièrement des messages de contrôle pour mettre à jour les tables de routage. Cette approche garantit qu'une route vers chaque destination est disponible immédiatement lorsqu'un paquet doit être envoyé. Les tables de routage sont ajustées à chaque changement de la topologie du réseau. Les deux principales méthodes utilisées sont l'état des liaisons et le vecteur de distance[26]

#### Avantage et inconvénients

Avec un protocole de routage proactif, les routes sont immédiatement disponibles lorsque nécessaires, ce qui élimine le délai pour les trouver. Théoriquement, cette approche présente peu de problèmes, sauf que les routes peuvent changer plus fréquemment que les demandes de route elles-mêmes. Dans de tels cas, le trafic des messages de contrôle et de mise à jour des tables de routage est élevé. De plus, ce trafic est en partie inutile car seules quelques-unes des routes seront réellement utilisées par les applications. Cela entraîne un gaspillage de la capacité du réseau sans fil en termes de bande passante.

#### 1.4.4.2 Protocole de routage réactif

Ces protocoles reposent sur la découverte et le maintien des routes. Lorsqu'une communication est nécessaire, une procédure de découverte globale des routes est déclenchée. Cette procédure se termine soit lorsque la route est trouvée, soit lorsque toutes les possibilités ont été examinées. Une fois la route établie, elle est maintenue tant que la destination reste accessible ou jusqu'à ce qu'elle ne soit plus nécessaire. Les protocoles basés sur ce principe incluent DSR, AODV, TORA, ABR, et RDMAR.[26]

### Avantage et inconvénients

Contrairement aux protocoles proactifs, les protocoles réactifs n'envoient aucun message de contrôle pour des routes qui ne sont pas utilisées. Cette approche permet d'éviter le gaspillage des ressources du réseau, ce qui se traduit par des économies de bande passante et d'énergie. Mais la mise en place d'une route par inondation peut être coûteuse et provoquer des délais importants avant l'ouverture de la route

#### 1.4.4.3 Protocole de routage hybride :

Un protocole de routage hybride appartient à une catégorie de protocoles réseau qui fusionne les caractéristiques des protocoles précédentes. Ces protocoles combinent l'utilisation de vecteurs de distance pour les communications entre routeurs voisins avec une représentation globale de la topologie du réseau, incluant des informations d'état de liaison. Cette approche permet de prendre des décisions de routage plus efficaces.[2]

#### 1.4.5 Problème de routage dans les réseaux ad hoc

Dans les réseaux ad hoc, le routage est confronté à un défi majeur : la topologie du réseau change constamment en raison de la mobilité des nœuds. Contrairement aux réseaux filaires traditionnels.

1. **Ruptures fréquentes des liens** : Les nœuds se déplacent librement, ce qui entraîne des changements fréquents dans les connexions sans fil entre les nœuds. Ces ruptures de liens rendent les routes établies rapidement obsolètes et invalident les tables de routage.
2. **Convergence lente des algorithmes** : Lorsque la topologie change constamment, les algorithmes de routage doivent converger rapidement pour refléter la nouvelle topologie. Cependant, certains algorithmes souffrent d'une convergence lente, ce qui peut entraîner des boucles de routage temporaires et une connectivité dégradée.
3. **Signalisation de contrôle excessive** : Pour maintenir des routes à jour, les nœuds doivent échanger des messages de contrôle. Dans un environnement dynamique, cette signalisation peut devenir excessive, consommant une partie significative de la bande passante et de l'énergie disponibles.

4. **Connectivité intermittente** : En raison de la mobilité des nœuds, des liens peuvent être temporairement rompus, rendant certaines destinations inaccessibles pendant un certain temps. Les algorithmes de routage doivent pouvoir gérer cette connectivité intermittente.
5. **Qualité de service dégradée** : Les changements fréquents de topologie peuvent entraîner des retards supplémentaires, une gigue élevée et des pertes de paquets, dégradant la qualité de service des communications.

Pour relever ces défis, différentes approches de routage ont été proposées. Le choix de l'algorithme de routage approprié dépend des caractéristiques spécifiques du réseau ad hoc, telles que la densité, la mobilité des nœuds et les exigences en termes de débit et de délai.

## 1.5 Concepts de centralité des nœuds

Dans les réseaux ad hoc, caractériser la topologie nécessite d'analyser la centralité des nœuds, une propriété clé. La centralité reflète l'importance relative d'un nœud au sein du réseau, quantifiée par différentes mesures telles que le degré, la proximité ou l'intermédiarité[17]. Les nœuds centraux jouent un rôle prépondérant dans le routage et la diffusion des données. Leur suppression, même en petit nombre, peut avoir un impact dramatique sur les performances globales, conformément au principe de Pareto, qui stipule que 80 % des effets sont souvent le résultat de 20 % des causes.

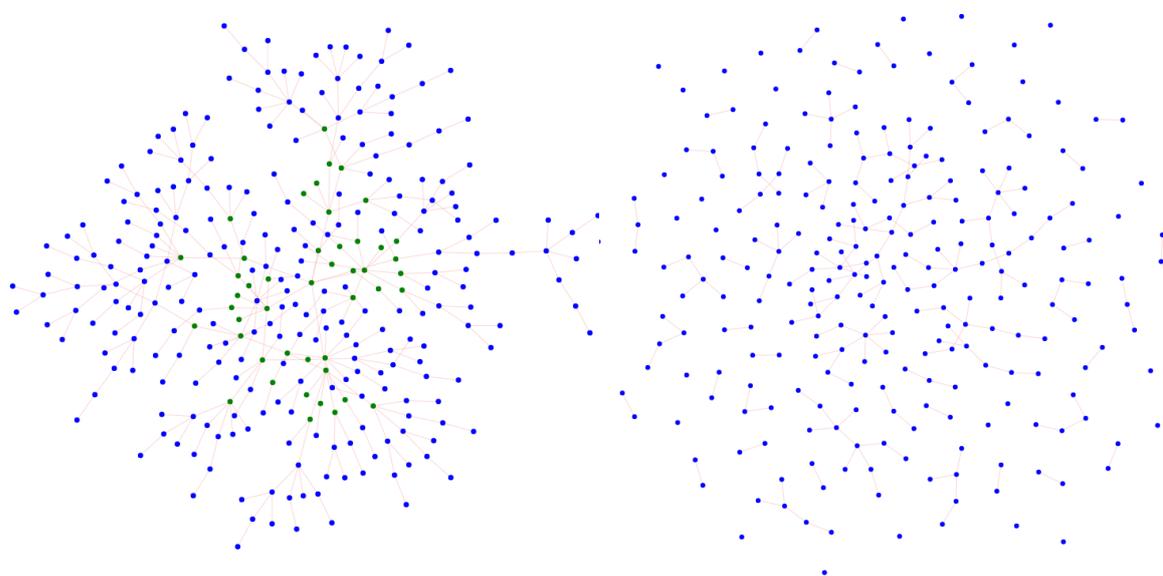


FIGURE 1.8 – Représentation d'un réseau ad hoc : à gauche, le réseau ad hoc avec toute les connexion ; à droite, le réseau après la suppression de 20% des nœuds.

Un nœud est considéré comme central s'il constitue le seul lien reliant certaines composantes du réseau qui seraient autrement isolées. Autrement dit, il existe un sous-ensemble de nœuds qui ne peuvent communiquer qu'en empruntant des chemins passant par ce nœud central, malgré les différentes notions de centralité qui restent des propriétés distinctes.

Il est donc crucial d'identifier précisément les nœuds centraux. Leur suppression, même en faible proportion, fragmente sévèrement le réseau en déconnectant de larges portions et en augmentant exponentiellement les distances de communication. Selon Dinh et al.[35], la suppression des nœuds avec la plus haute centralité d'intermédiarité

amène l'augmentation rapide du diamètre du réseau et de la longueur moyenne de chemin en ajoutant le plus de temps de commutation. Leur recherche sur des réseaux ad hoc de 500 nœuds, créés aléatoirement, montre que si seulement 10% des nœuds centraux sont enlevés, alors le diamètre est accru de 2,5 fois.

De même, Neumayer et al.[32] ont étudié l'effet de la suppression des nœuds centraux selon différentes métriques (degré, proximité, intermédiarité) sur la taille de la plus grande composante connexe dans un réseau ad hoc. Leurs résultats indiquent que cibler les nœuds à forte centralité d'intermédiarité ou de proximité fragmente le réseau de manière bien plus sévère que de s'attaquer aux nœuds à haut degré.

La détection préalable des nœuds centraux à l'aide de métriques et d'algorithmes spécialisés est essentielle pour développer des stratégies intelligentes de routage, de transfert de données, de placement de ressources et de reconfiguration topologique [17]. En effet, protéger et surveiller ces nœuds centraux, véritables piliers de la connectivité des réseaux ad hoc, est d'une importance cruciale. En concentrant nos efforts sur la protection et la surveillance de ces nœuds clés, nous pouvons garantir une connectivité résiliente face aux défaillances et à la mobilité constante des réseaux. En fin de compte, pour maintenir des performances optimales dans les réseaux ad hoc, il est indispensable d'identifier et de gérer efficacement ces nœuds centraux, tant pour le flux de trafic que pour l'intégrité de la topologie du réseau.

### 1.5.1 Mesures de centralité des nœuds

La centralité d'un nœud dans un réseau est importante vu son impact sur la connectivité et les flux d'informations. Plusieurs mesures ont été présentées pour quantifier cet aspect[36][33] :

1. **La centralité de degré (degree centrality)** est la mesure la plus simple. Elle correspond tout simplement au nombre de voisins directement connectés à un nœud donné. Un nœud ayant un degré élevé est susceptible d'être mieux informé et d'avoir une influence plus grande sur la diffusion de l'information.

La centralité d'un sommet  $v$  dans un graphe  $G = (V, E)$  avec  $N = |V|$  sommets et  $M = |E|$  arêtes est définie par :

$$C_{\text{deg}}(v) = \text{degré}(v)$$

2. **La centralité de proximité (closeness centrality)** évalue la distance moyenne (c'est-à-dire le nombre minimum de liens) entre un nœud et tous les autres

nœuds du réseau. Un nœud ayant une centralité de proximité élevée peut atteindre rapidement le reste du réseau et accéder plus efficacement à l'information qui y circule.

Formellement, la centralité de proximité d'un sommet  $v$  dans un graphe  $G = (V, E)$  avec  $N = |V|$  sommets et  $M = |E|$  arêtes est définie par :

$$C(v) = \frac{1}{\sum_u d(u, v)}$$

où  $d(u, v)$  représente la distance entre les sommets  $v$  et  $u$ .

3. **La centralité d'intermédierité (betweenness centrality)** mesure la fréquence à laquelle un nœud se trouve sur les chemins les plus courts reliant d'autres paires de nœuds. Plus cette valeur est élevée, plus le nœud sert d'intermédiaire pour l'échange d'informations et exerce un contrôle potentiel sur les flux de communication.

Formellement, la centralité d'intermédierité  $C_B(v)$  d'un nœud  $v$  est définie comme suit : l'intermédierité d'un sommet  $v$  dans un graphe  $G = (V, E)$  avec  $|V|$  sommets est calculée comme suit :

- (a) Pour chaque couple de sommets  $(s, t)$ , on calcule les plus courts chemins qui les relient.
- (b) Pour chaque couple  $(s, t)$ , on détermine la proportion de ces plus courts chemins qui passent par le sommet  $v$ .
- (c) On somme cette proportion sur tous les couples  $(s, t)$  de sommets.

De façon plus concise, l'intermédierité peut être représentée par :

$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

où  $\sigma_{st}$  est le nombre total de plus courts chemins du sommet  $s$  au sommet  $t$ , et  $\sigma_{st}(v)$  est le nombre de ces chemins qui passent par  $v$ .

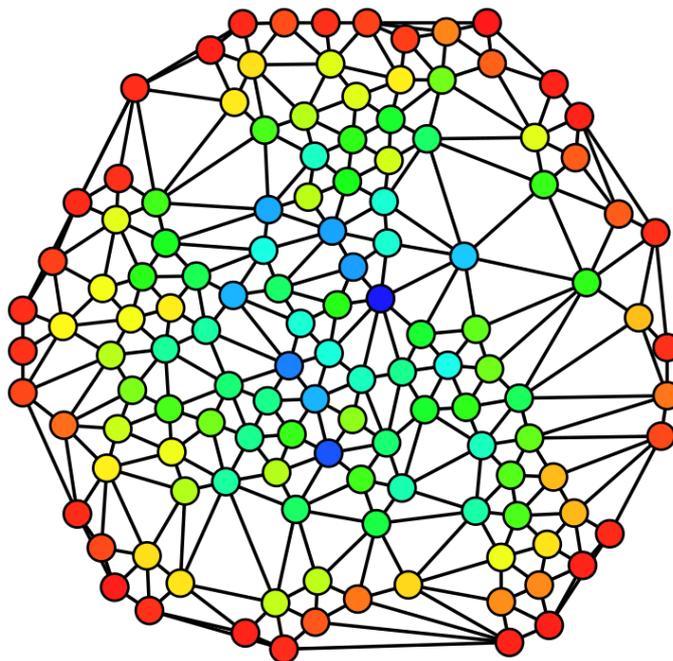


FIGURE 1.9 – Centralité d’intermédiation des nœuds (La teinte, allant de rouge = 0 à bleu = max, indique l’intermédiation des nœuds)

4. **L’efficacité de diffusion d’information (information diffusion efficiency)** évalue la vitesse et la portée avec laquelle l’information se propage à partir d’un nœud donné vers le reste du réseau. Elle tient compte des différentes longueurs de chemins et de la redondance des connexions.
5. **La centralité de vecteur propre (eigenvector centrality)** : La centralité de vecteur propre (eigenvector centrality) ne prend pas seulement en compte le nombre de connexions d’un nœud, mais également l’importance relative de ces connexions. Cette centralité attribue une pondération à chaque nœud en fonction des connexions qu’il a avec d’autres nœuds centraux. Un nœud connecté à des nœuds importants aura ainsi une centralité d’eigenvecteur plus élevée.

Cette centralité est calculée en résolvant l’équation suivante, exprimée en notation vectorielle comme une équation au vecteur propre :

$$\mathbf{Ax} = \lambda \mathbf{x}$$

où  $\mathbf{A}$  est la matrice d’adjacence du graphe,  $\mathbf{x}$  est le vecteur des centralités d’eigenvecteur, et  $\lambda$  est une constante (valeur propre) associée.

### 1.5.2 Algorithmes dédiés à la détection des nœuds centraux

Bien que les mesures de centralité fournissent des indications précieuses, des algorithmes spécifiques ont été développés pour identifier de manière fiable ces nœuds.

#### 1.5.2.1 Approches par composantes bi-connexes

Les approches par composantes bi-connexes reposent sur la théorie des graphes et la notion de connectivité. Un graphe est dit bi-connexé s'il ne possède aucun nœud central.

L'idée principale est de décomposer le réseau en ses composantes bi-connexes maximales, qui représentent les sous-graphes bi-connexes de plus grande taille. Les nœuds centraux du réseau sont alors les nœuds reliant ces différentes composantes bi-connexes. Leur suppression provoquerait une déconnexion du réseau.

L'algorithme de Hopcroft et Tarjan [16] est une référence dans ce domaine. Il identifie de manière efficace les composantes bi-connexes et les points d'articulation avec une complexité temporelle linéaire  $O(|V| + |E|)$ , où  $|V|$  et  $|E|$  représentent respectivement le nombre de nœuds et d'arêtes du graphe. Cet algorithme fonctionne en deux phases :

Une phase de parcours en profondeur du graphe pour assigner des numéros d'ordre aux nœuds et détecter les points d'articulation potentiels. Une phase de traitement des points d'articulation potentiels pour déterminer s'ils sont effectivement des points d'articulation et identifier les composantes bi-connexes.

#### 1.5.2.2 Approches spectrales (valeurs propres de la matrice Laplacienne)

Les approches spectrales exploitent les propriétés algébriques de la matrice Laplacienne du graphe, qui encode les informations topologiques du réseau. Cette matrice, notée  $L$ , est définie à partir de la matrice d'adjacence  $A$  du graphe et de la matrice diagonale  $D$  des degrés des nœuds, selon l'équation suivante [12] :

$$L = D - A$$

Où :

$A$  est la matrice d'adjacence du graphe, une matrice carrée de taille  $n \times n$  ( $n$  étant le nombre de nœuds) avec des entrées  $A(i, j) = 1$  si les nœuds  $i$  et  $j$  sont connectés, et 0 sinon.  $D$  est la matrice diagonale des degrés des nœuds, une matrice diagonale de

taille  $n \times n$  avec  $D(i, i) = \deg(i)$ , le degré du nœud  $i$  (c'est-à-dire le nombre de voisins du nœud  $i$ ).

L'analyse des valeurs propres et vecteurs propres de cette matrice Laplacienne permet d'extraire des informations pertinentes sur la structure et la connectivité du réseau. Plus précisément, les vecteurs propres associés aux petites valeurs propres non nulles de  $L$  mettent en évidence les régions du réseau qui sont faiblement connectées au reste du graphe. Mathématiquement, si  $\lambda$  est une valeur propre de  $L$  et  $x$  est le vecteur propre associé, alors :

$$Lx = \lambda x$$

Les nœuds ayant des coefficients élevés (en valeur absolue) dans ces vecteurs propres  $x$  correspondant aux petites valeurs propres non nulles  $\lambda$  sont souvent les nœuds centraux reliant ces régions faiblement connectées. Des métriques comme la valeur propre algébrique [9] ou le vecteur propre creux [25] permettent d'identifier et de quantifier la criticité des nœuds à partir de cette analyse spectrale. Ces approches offrent l'avantage d'être applicables à des réseaux de grande taille et pondérés, contrairement aux algorithmes par composantes bi-connexes qui sont limités aux graphes non pondérés de taille modérée.

## 1.6 Conclusion

Les réseaux ad hoc, caractérisés par leur nature dynamique et décentralisée, présentent des défis uniques en termes de gestion et d'optimisation. Le concept de centralité des nœuds s'avère crucial dans ce contexte, offrant un moyen d'identifier les éléments clés du réseau. Cette approche permet de mieux comprendre la structure et le fonctionnement des réseaux ad hoc, ouvrant ainsi des perspectives pour améliorer leur efficacité, leur robustesse et leur adaptabilité dans divers scénarios d'application. L'étude de la centralité des nœuds dans les réseaux ad hoc reste un domaine de recherche prometteur, avec des implications importantes pour le développement futur des technologies de communication sans fil.

## 1.7 Problématique et positionnement de la thèse

Cette thèse se positionne dans le cadre de la recherche avancée sur les réseaux ad hoc, des réseaux sans fil où les appareils peuvent se connecter directement les uns aux autres sans nécessiter une infrastructure fixe préétablie, comme des stations de base ou des routeurs. Ces réseaux jouent un rôle de plus en plus important dans divers domaines, tels que les interventions d'urgence où une infrastructure traditionnelle peut être absente, les opérations militaires nécessitant une communication robuste et flexible sur le terrain, ou encore dans les environnements industriels où la connectivité doit s'adapter rapidement aux changements dans les processus de fabrication.

Cependant, la gestion efficace de ces réseaux pose des défis uniques en raison de leur nature décentralisée et dynamique. L'un des défis majeurs est l'identification des noeuds centraux, c'est-à-dire les appareils qui sont essentiels pour maintenir une connectivité efficace et une transmission de données fluide dans le réseau. Cette tâche est complexe en raison de la mobilité des appareils, des changements fréquents dans la topologie du réseau et des variations dans les connexions entre les appareils.

La thèse se concentre donc sur la résolution de cette problématique fondamentale. Elle vise à développer des méthodes innovantes et des algorithmes efficaces pour identifier les noeuds centraux dans les réseaux ad hoc, malgré leur nature dynamique et complexe. En comprenant mieux comment ces noeuds centraux contribuent à la robustesse et à la performance globale du réseau, il sera possible de concevoir des stratégies de gestion plus efficaces et de renforcer ainsi l'utilité et la fiabilité des réseaux ad hoc dans divers contextes d'application.

Pour aborder cette problématique, nous proposons la formulation mathématique suivante :

Soit  $G = (V, E)$  un graphe représentant le réseau ad hoc, où :

- $V$  est l'ensemble des noeuds
- $E$  est l'ensemble des arêtes entre les noeuds

Définissons  $C(v)$  comme une fonction de centralité quantifiant l'importance du noeud  $v \in V$ .

**Objectif :** Trouver un sous-ensemble  $S \subseteq V$  qui maximise l'efficacité du réseau :

$$\max_S \Phi(S)$$

sous les contraintes :

$$Q(S) \geq Q_{min}$$

$$\forall v \in S, C(v) \geq \beta$$

où :

- $\Phi(S)$  est une fonction d'efficacité du réseau
- $Q(S)$  est une fonction mesurant la qualité globale de la couverture du réseau par les nœuds centraux  $S$
- $Q_{min}$  est un seuil minimal de qualité de couverture
- $\beta$  est un seuil minimal de centralité

Cette formulation vise à développer des méthodes innovantes pour identifier les nœuds centraux dans les réseaux ad hoc, contribuant ainsi à améliorer leur robustesse et leur performance globale dans divers contextes d'application.

# Méthodes approchées : heuristiques et métaheuristiques

---

## 2.1 Introduction

Dans ce chapitre, nous allons nous pencher sur les méthodes approchées en optimisation, des outils essentiels pour s'attaquer à des problèmes complexes. Notre attention se portera plus sur les heuristiques et les métaheuristiques.

Nous commencerons par explorer les heuristiques, ces techniques qui cherchent des solutions acceptables rapidement, sans garantir l'optimalité. Ensuite, nous nous intéresserons aux métaheuristiques comme les algorithmes génétiques ou l'optimisation par colonies de fourmis, qui offrent des stratégies plus élaborées.

L'objectif est de comprendre ces approches, leurs forces et leurs limites, pour mieux les appliquer à divers problèmes d'optimisation. Ce tour d'horizon nous préparera à aborder des défis spécifiques dans la suite de nos travaux.

## 2.2 Classification des approches d'optimisation

Les problèmes d'optimisation complexes rencontrés dans divers domaines nécessitent des approches méthodologiques adaptées. Ces approches peuvent être classées en plusieurs catégories principales, chacune présentant des avantages et des inconvénients selon la nature du problème et les contraintes de résolution. Voici les principales catégories d'approches d'optimisation :

- Méthodes exactes

- Méthodes approchées : Les heuristiques et Les métaheuristiques.
- Méthodes d'amélioration.
- Méthodes à deux phases.

### 2.2.1 Méthodes exactes

Les méthodes exactes (aussi appelées méthodes d'optimisation exacte) sont des algorithmes qui garantissent de trouver la solution optimale d'un problème d'optimisation combinatoire en un temps fini, à condition que le problème soit résoluble dans les limites de capacité de calcul disponibles. Ces méthodes explorent systématiquement tout l'espace de recherche des solutions possibles jusqu'à ce que la solution optimale soit identifiée[3].

### 2.2.2 Méthodes approchées

Les méthodes approchées, également connues sous le nom d'algorithmes d'approximation, regroupent les heuristiques et les métaheuristiques. Ce sont des approches algorithmiques visant à trouver une solution réalisable de qualité acceptable pour un problème d'optimisation combinatoire. Contrairement aux méthodes exactes, elles ne garantissent pas l'optimalité, mais offrent des temps de calcul raisonnables. Plutôt que d'explorer exhaustivement tout l'espace de recherche, ces méthodes approchées utilisent des stratégies intelligentes pour générer des solutions de qualité satisfaisante dans des délais acceptables[20].

## 2.3 Heuristique

Le mot heuristique, dérivé de la langue grec, vient du verbe heuriskein qui signifie "trouver". Une heuristique représente une approche offrant une solution approximative à un problème d'optimisation complexe, sans garantie de fournir la solution optimale. Ce type de méthode s'appuie souvent sur des règles empiriques ou des stratégies intelligentes pour naviguer efficacement dans l'espace de recherche et trouver rapidement une solution satisfaisante sans nécessiter une exploration exhaustive de toutes les possibilités[37].

## 2.4 Métaheuristique

Les métaheuristiques sont des algorithmes d'optimisation avancés et génériques qui combinent de façon ingénieuse diverses stratégies afin de guider efficacement la recherche vers l'obtention de solutions quasi-optimales. Contrairement aux heuristiques spécifiques à un problème, elles sont des approches générales applicables à de nombreux problèmes d'optimisation complexes issus de différents domaines.

Ces méthodes itératives orchestrent judicieusement plusieurs composants algorithmiques et opérateurs de recherche complémentaires, tels que l'intensification locale et la diversification exploratoire. Cette hybridation contrôlée de mécanismes variés confère aux métaheuristiques une grande puissance pour résoudre des problèmes ardu.

Si leur sophistication implique une complexité supérieure aux heuristiques élémentaires, les métaheuristiques permettent d'atteindre des performances très élevées. Elles constituent ainsi des outils incontournables pour relever les défis d'optimisation les plus difficiles[14].

La plupart des métaheuristiques tirent leur inspiration des systèmes naturels. Par exemple, le recuit simulé s'inspire d'un processus métallurgique, les algorithmes évolutionnaires et génétiques sont basés sur les principes de l'évolution darwinienne et de la biologie, tandis que la recherche tabou s'inspire de la mémoire humaine.

Les métaheuristiques sont généralement divisées en deux catégories : celles basées sur une solution unique et celles basées sur une population de solutions. Dans cette section, nous présentons quelques exemples de métaheuristiques appartenant à ces deux catégories.

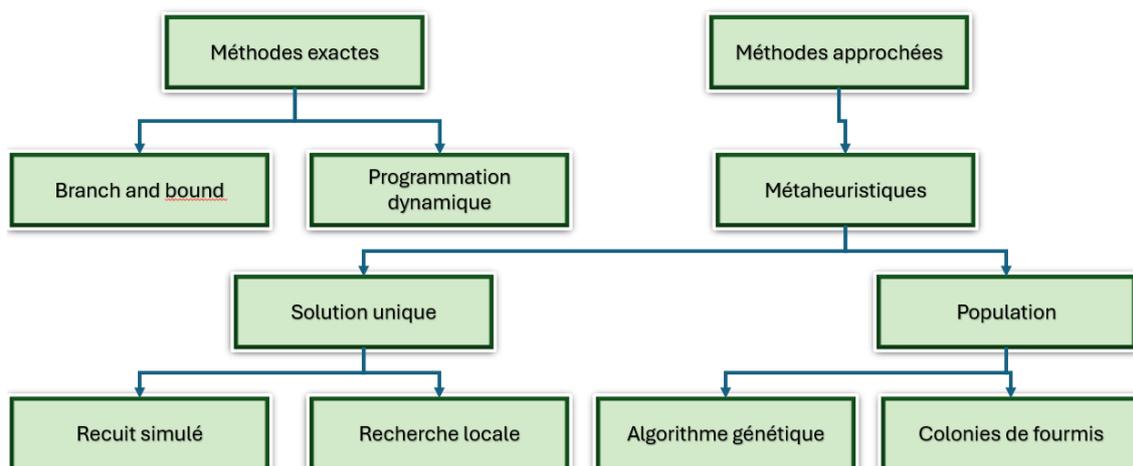


FIGURE 2.1 – Classification des métaheuristiques.

### 2.4.1 Les métaheuristiques à base de solution unique

Les métaheuristiques à base de solution unique s'appuient sur le concept de voisinage pour améliorer la qualité de la solution actuelle. Elles commencent la recherche avec une seule solution initiale, puis explorent l'espace de recherche en générant et en améliorant itérativement cette solution. La qualité de la solution finale est fortement influencée par les ajustements réalisés par les opérateurs de voisinage. En effet, des transformations inappropriées de la solution initiale peuvent conduire la recherche vers un optimum local dans un voisinage donné, ce qui peut bloquer la recherche en aboutissant à une solution de qualité insatisfaisante[30].

De nombreuses métaheuristiques à base de solution unique ont été proposées dans la littérature. Parmi lesquelles : la descente, le recuit simulé, la recherche tabou, la recherche à voisinage variable, la recherche locale, etc.

#### 2.4.1.1 Le recuit simulé

L'algorithme du recuit simulé a été proposé par Kirkpatrick, Gelatt et Vecchi, tirant son inspiration de la procédure de refroidissement contrôlé utilisée en métallurgie. Les métallurgistes chauffent le métal à blanc, puis le laissent refroidir très lentement afin d'obtenir un alliage sans défauts. Une diminution brutale de la température provoque la formation d'une structure amorphe, correspondant à un optimum local. En revanche, un refroidissement progressif permet d'atteindre une structure bien ordonnée, équiva-

lente à l'optimum global. Le fonctionnement de cette métaheuristique s'appuie sur l'algorithme de Metropolis, datant de 1953. Son principe, dans le cadre d'un problème de maximisation, peut se résumer ainsi :

---

**Algorithm 1** Recuit simulé[30]

---

- 1: Partir d'une solution initiale  $s$
  - 2: Initialiser une température  $T$  à une valeur élevée
  - 3: Calculer la valeur de la fonction objectif  $f(s)$  pour la solution  $s$
  - 4: Générer une solution voisine  $s'$  de  $s$
  - 5: Calculer  $f(s')$
  - 6: Calculer l'écart d'évaluation  $\Delta f = f(s') - f(s)$
  - 7: **if**  $\Delta f > 0$  **then**
  - 8:     Remplacer  $s$  par  $s'$
  - 9: **else**
  - 10:    Générer un nombre aléatoire  $r$  entre 0 et 1
  - 11:    **if**  $r < \exp\left(\frac{\Delta f}{T}\right)$  **then**
  - 12:     Remplacer  $s$  par  $s'$
  - 13:    **end if**
  - 14: **end if**
- 

Le recuit simulé permet d'accepter, avec une certaine probabilité, des solutions de moins bonne qualité que la solution courante. Cela vise à diversifier la recherche et éviter de rester bloqué dans un optimum local. Une solution globale optimale peut en effet se trouver dans le voisinage d'une solution de qualité médiocre, et non d'une bonne solution locale.

Toutefois, l'acceptation aveugle de solutions dégradées risque de faire perdre la trace des meilleures solutions trouvées précédemment. Il est donc recommandé de conserver une variable stockant la meilleure solution rencontrée au cours de la recherche. La probabilité d'accepter une solution de moindre qualité dépend de deux facteurs : l'écart d'évaluation  $\Delta f$  et la température  $T$ . Plus  $T$  est élevé, plus cette probabilité est grande. La valeur de  $T$  décroît progressivement au fil des itérations, pour tendre vers 0. Un refroidissement trop rapide peut mener à une convergence prématurée vers un optimum local médiocre, tandis qu'un refroidissement trop lent induit des temps de calcul excessifs. Le réglage de la température initiale et de son schéma de décroissance impacte fortement les performances de l'algorithme[30].

Le recuit simulé, facile à mettre en œuvre et à adapter à de nombreux problèmes, offre l'avantage de pouvoir s'extraire des optimums locaux. Cependant, son paramétrage requiert un réglage délicat, et sa lenteur peut être problématique sur les instances de grande taille.

subsubsectionLa recherche locale simple La recherche locale, également connue sous le nom de Descente, est l'un des algorithmes d'optimisation les plus anciens. Elle démarre à partir d'une solution initiale donnée et explore itérativement le voisinage de cette solution pour améliorer sa qualité. À chaque itération, l'algorithme remplace la solution actuelle par un voisin qui améliore la fonction objective. Lorsque tous les voisins candidats sont moins bons que la solution actuelle, cela signifie qu'un optimum local est atteint et que la recherche s'arrête. Les variantes de la recherche locale incluent la descente déterministe, la descente stochastique et la descente vers le premier meilleur voisin. Bien que simple et rapide, cette méthode peut rester bloquée au premier optimum local rencontré, qui n'est pas nécessairement l'optimum global du problème. L'objectif principal de la recherche locale est d'accélérer la recherche en restreignant le voisinage exploré. Les différentes variantes de la recherche locale se distinguent par l'ordre dans lequel les solutions voisines sont générées (déterministe/stochastique) et par la stratégie de sélection de la solution voisine [10].

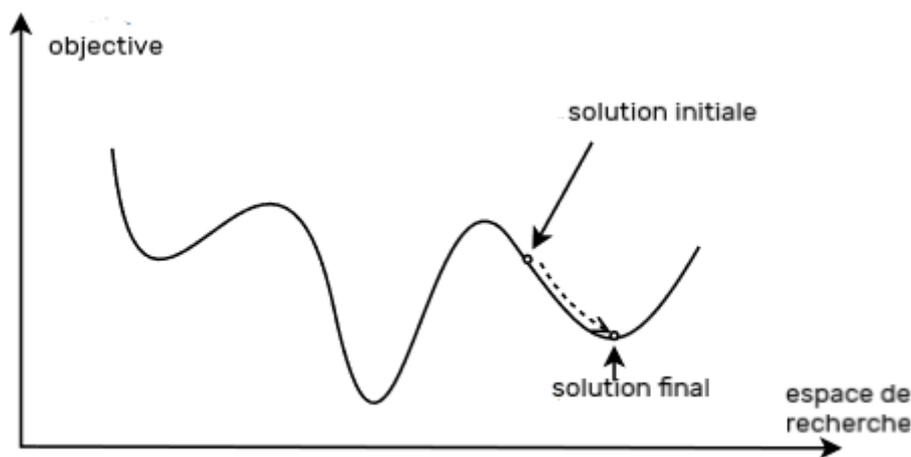


FIGURE 2.2 – Comportement de la recherche locale (descente la plus raide) dans un paysage donné.)[10]

La recherche locale est une méthode d'optimisation qui explore l'espace de recherche en se déplaçant d'une solution à une solution voisine de meilleure qualité. On peut représenter cet espace de recherche comme un graphe  $G = (X, V)$ , où  $X$  est l'ensemble

des solutions réalisables et  $V$  représente la relation de voisinage entre les solutions. Dans ce graphe, chaque sommet correspond à une solution réalisable, et un arc reliant deux sommets indique qu'ils sont des solutions voisines. L'algorithme de recherche locale fonctionne de la manière suivante :

---

**Algorithm 2** Algorithme de recherche locale[10]

---

- 1: Générer une solution initiale  $s_0 \in X$
  - 2: **repeat**
  - 3:   Générer  $N(s_i)$  l'ensemble des solutions voisines de  $s_i$
  - 4:   **if**  $\exists s' \in N(s_i)$  telle que  $f(s') < f(s_i)$  **then**
  - 5:     Choisir  $s_{i+1} \in N(s_i)$  tel que  $f(s_{i+1}) = \min_{s' \in N(s_i)} f(s')$
  - 6:   **else**
  - 7:      $s_i$  est un optimum local, fin de l'algorithme
  - 8:   **end if**
  - 9: **until** Critère d'arrêt atteint
  - 10: Retourner  $s_i$  (optimum local)
- 

L'algorithme commence par générer une solution initiale  $s_0 \in X$ . Ensuite, il entre dans une boucle itérative où, à chaque itération  $i$ , il génère l'ensemble  $N(s_i)$  des solutions voisines de la solution courante  $s_i$ . Parmi ces solutions voisines, il sélectionne la meilleure solution  $s_{i+1}$  selon un critère d'optimisation (minimisation dans ce cas), et remplace la solution courante  $s_i$  par  $s_{i+1}$ . Ce processus se poursuit jusqu'à ce qu'un critère d'arrêt soit satisfait, comme atteindre un nombre maximal d'itérations ou ne plus trouver de meilleure solution voisine (ce qui indique que l'algorithme a convergé vers un optimum local). Quelques caractéristiques supplémentaires :

La taille de la séquence de solutions générées  $(s_0, s_1, s_2, \dots, s_k)$  est inconnue a priori. Chaque solution  $s_{i+1}$  est un voisin de  $s_i$  ( $s_{i+1} \in N(s_i)$ ). La fonction  $f(s_{i+1}) < f(s_i)$  pour tout  $i$  dans  $[0, k-1]$ , ce qui signifie que l'algorithme progresse vers des solutions de meilleure qualité à chaque itération.  $s_k$  est un optimum local, c'est-à-dire qu'il n'existe aucune solution voisine de meilleure qualité.

En plus de la définition de la solution initiale et du voisinage, la conception d'un algorithme de recherche locale nécessite de définir une stratégie de sélection du voisin qui déterminera la prochaine solution courante.

Plusieurs stratégies peuvent être appliquées pour la sélection d'un meilleur voisin dans les algorithmes de recherche locale :[10]

- **La meilleure amélioration (descente la plus raide)** : Avec cette stratégie, le meilleur voisin, c'est-à-dire celui qui améliore le plus la fonction objectif, est sélectionné. Le voisinage est exploré de manière exhaustive et tous les mouvements possibles sont essayés pour une solution afin de sélectionner le meilleur voisin. Cette stratégie peut devenir coûteuse en temps de calcul pour les grands voisinages.
- **La première amélioration** : Cette stratégie consiste à choisir le premier voisin améliorant par rapport à la solution courante. Dès qu'un voisin améliorant est trouvé, il est immédiatement sélectionné pour remplacer la solution actuelle. Cette approche implique une exploration partielle du voisinage, où l'évaluation des voisins s'arrête dès qu'une amélioration est constatée. Elle suit généralement un ordre prédéfini pour la génération des voisins.
- **La sélection aléatoire** : Dans cette stratégie, un voisin est choisi au hasard parmi ceux de la solution courante.

Un compromis en termes de qualité des solutions et de temps de recherche peut être envisagé en utilisant la stratégie de la première amélioration lorsque la solution initiale est générée aléatoirement.

### 2.4.2 Les métaheuristiques à base de population de solutions

Les métaheuristiques à base de population de solutions sont des approches algorithmiques qui s'inspirent du comportement des populations naturelles pour résoudre des problèmes complexes d'optimisation. Ces techniques s'appuient sur la notion de collaboration entre individus au sein d'une population, en commençant leur exploration avec un ensemble varié de solutions et en identifiant la meilleure solution possible parmi cet ensemble, représentant ainsi l'optimum global du problème. Cette approche, qui exploite la diversité des solutions, augmente les chances de découvrir des solutions de qualité. Un large éventail de méthodes reposant sur l'utilisation de populations de solutions a été étudié dans la littérature, allant des algorithmes évolutionnaires aux techniques basées sur l'intelligence collective comme les essaims de particules et les colonies de fourmis, ayant fait l'objet d'une attention particulière au cours des dernières décennies[30].

Dans ce contexte, il est essentiel de comprendre le fonctionnement de ces métaheuristiques et leur potentiel à résoudre une grande variété de problèmes.

### 2.4.2.1 Les algorithmes génétiques

L'algorithme génétique est une méthode de résolution de problèmes puissante, conceptualisée par John Holland en 1975. Inspirée des principes de la génétique et de l'évolution naturelle décrits par Charles Darwin, elle imite les processus biologiques tels que la reproduction et la sélection naturelle afin de trouver des solutions optimales à des problèmes complexes. Cette approche révolutionnaire utilise des concepts analogues à ceux de la biologie comme les gènes, les chromosomes, les individus, les populations et les générations pour explorer efficacement l'espace des solutions possibles[8].

Un gène représente l'unité de base encodant une caractéristique ou un aspect spécifique de la solution, pouvant être représenté par un bit, un entier, un nombre réel ou un caractère. Un chromosome est une séquence ordonnée de gènes formant une solution potentielle au problème. Un individu est une entité composée d'un ou plusieurs chromosomes, encodant ainsi une solution candidate complète. Une population est un ensemble regroupant de multiples individus distincts, chacun représentant une solution potentielle différente. Enfin, une génération constitue une étape de l'évolution au cours de laquelle des opérations sont appliquées sur la population actuelle pour produire une nouvelle population pour la génération suivante.

Le déroulement typique d'un algorithme génétique comprend les étapes suivantes à chaque génération[24] :

1. **Évaluation** : Chaque individu de la population est évalué selon une fonction objectif qui mesure sa performance ou son aptitude (fitness) à résoudre le problème.
2. **Sélection** : Les meilleurs individus, ayant les meilleures valeurs de fitness, sont sélectionnés préférentiellement pour la reproduction, en utilisant diverses techniques de sélection.
3. **Reproduction** : Les individus sélectionnés subissent des opérations de croisement (ou recombinaison) et de mutation. Le croisement consiste à combiner les gènes de deux individus parents pour générer de nouveaux individus enfants, potentiellement plus performants. La mutation introduit une variation aléatoire dans les gènes d'un individu afin de maintenir une diversité génétique au sein de la population.
4. **Remplacement** : Une partie ou la totalité des nouveaux individus enfants générés remplacent les individus les moins performants de la population actuelle.

Ce cycle itératif, inspiré du processus de sélection naturelle, permet à la population d'évoluer progressivement vers de meilleures solutions au fil des générations, jusqu'à ce qu'un critère d'arrêt prédéfini soit atteint, comme un nombre maximal de générations ou une valeur de fitness satisfaisante.

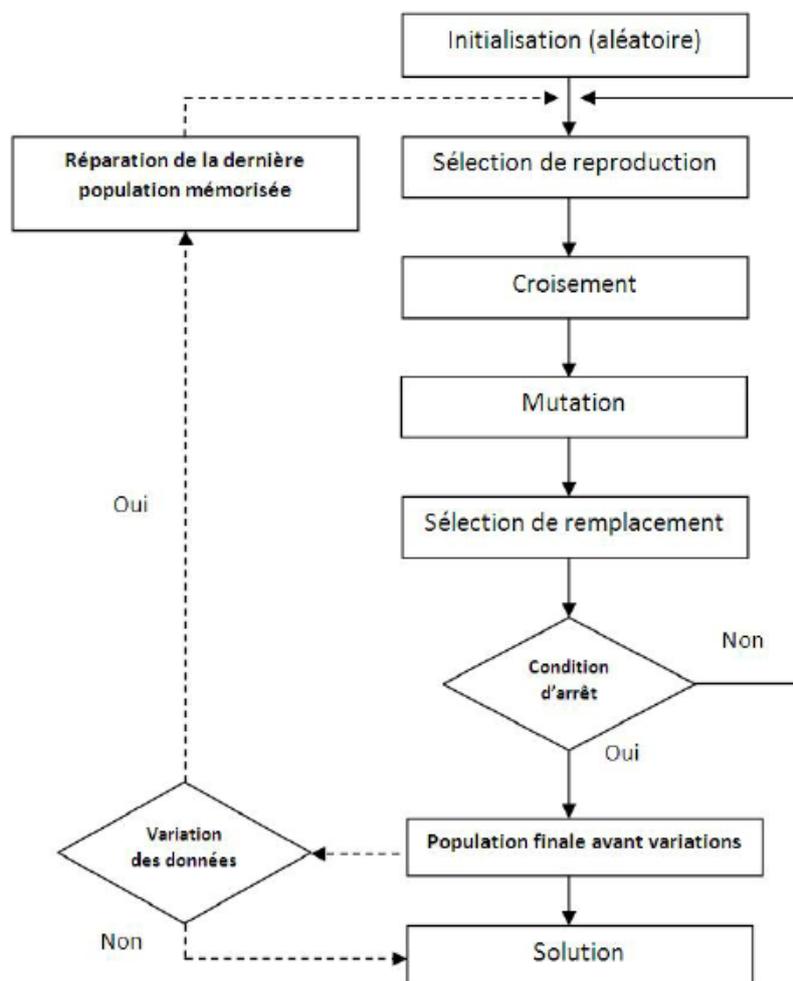


FIGURE 2.3 – Démarche d'un algorithme génétique. [6]

Outre les opérateurs de sélection, croisement, mutation et évaluation, les algorithmes génétiques nécessitent également d'autres opérateurs tels que le codage des individus et l'initialisation de la population. Les paramètres clés à régler sont la taille de la population, les probabilités de croisement et de mutation, ainsi que le nombre maximal de générations autorisé.

Grâce à leur capacité à simuler des mécanismes évolutifs, les algorithmes génétiques offrent un cadre flexible et robuste pour résoudre une grande variété de problèmes complexes, allant de l'optimisation combinatoire à la conception de systèmes intelligents. Leur utilisation couvre de vastes domaines de recherche, et leur potentiel d'application

continue d'être exploré dans divers champs scientifiques et technologiques. Voici une représentation algorithmique du processus d'un algorithme génétique :

---

**Algorithm 3** Algorithme Génétique[24]

---

- 1: Initialiser les paramètres nécessaires (taille population, taux croisement/mutation, critère d'arrêt, etc.)
  - 2: Initialiser une population de  $N$  individus (solutions candidates)
  - 3: Evaluer les  $N$  individus en utilisant la fonction objectif (fitness)
  - 4: **while** la condition d'arrêt n'est pas satisfaite **do**
  - 5: Utiliser un opérateur de sélection pour choisir  $K$  individus de la population, en favorisant les meilleurs
  - 6: Appliquer l'opérateur de croisement sur les  $K$  individus sélectionnés avec une probabilité  $P_c$ , pour générer de nouveaux individus enfants
  - 7: Appliquer l'opérateur de mutation sur les individus enfants avec une probabilité  $P_m$ , pour introduire de la diversité
  - 8: Utiliser l'opérateur d'évaluation pour calculer la fitness des nouveaux individus enfants
  - 9: Utiliser un opérateur de remplacement pour remplacer certains individus parents de la population par les nouveaux individus enfants
  - 10: **end while**
  - 11: Retourner la ou les meilleures solutions trouvées dans la population finale
- 

Cet algorithme présente les étapes fondamentales d'un algorithme génétique standard. Les détails des opérateurs utilisés (sélection, croisement, mutation, remplacement) peuvent varier selon les implémentations spécifiques[24].

#### 2.4.2.2 Les algorithmes de colonies de fourmis (Ant Colony Optimization : ACO)

L'algorithme de colonies de fourmis est une approche bio-inspirée appartenant à la famille des algorithmes d'intelligence en essaim. Introduit au début des années 90 par le trinôme Coloni, Dorigo et Maniezzo, il s'inspire du comportement collectif des fourmis lors de leur quête à la recherche de nourriture. Malgré leurs capacités individuelles limitées, ces insectes coopèrent de manière efficace pour résoudre des tâches complexes grâce à une communication indirecte via des phéromones.

Lorsqu'une fourmi part explorer son environnement à la recherche de nourriture, elle dépose une piste de phéromones sur son chemin. Au retour, cette piste chimique attire

les congénères vers le même trajet. Cependant, comme les phéromones s'évaporent avec le temps, les chemins les plus longs finissent par être délaissés au profit des plus courts où la concentration est élevée. Ce mécanisme d'auto-organisation permet aux fourmis de converger collectivement vers les trajets optimaux.

L'algorithme de colonies de fourmis reproduit ce principe pour résoudre des problèmes d'optimisation complexes. Chaque solution potentielle au problème est représentée par une fourmi virtuelle. Ces fourmis construisent progressivement leurs solutions en se déplaçant sur un graphe de décision et en déposant des quantités de phéromones virtuelles proportionnelles à la qualité des solutions trouvées. Les prochaines fourmis sont alors attirées préférentiellement vers les meilleurs chemins, permettant d'intensifier la recherche dans les régions prometteuses[4].

---

**Algorithm 4** Algorithme de Colonies de Fourmis[4]

---

- 1: Générer une population initiale de  $m$  fourmis artificielles (solutions candidates)
  - 2: Évaluer la qualité des  $m$  fourmis initiales selon un critère prédéfini
  - 3: **while** le critère d'arrêt n'est pas satisfait **do**
  - 4:   **for** chaque fourmi  $i$  de 1 à  $m$  **do**
  - 5:     Construire de manière itérative la solution de la fourmi  $i$
  - 6:     Déposer une quantité de phéromones virtuelles le long de la solution, proportionnelle à sa qualité
  - 7:   **end for**
  - 8:   Mettre à jour les niveaux de phéromones sur l'ensemble des solutions en appliquant une règle d'évaporation
  - 9:   Évaluer la qualité des  $m$  nouvelles solutions générées
  - 10: **end while**
  - 11: **return** la ou les meilleures solutions trouvées
- 

Le processus général comprend trois phases principales : la construction des solutions par les fourmis, le dépôt de phéromones sur les solutions, et l'évaporation des pistes de phéromones pour éviter une convergence prématurée. Ces étapes sont répétées de manière itérative jusqu'à ce qu'un critère d'arrêt soit satisfait, menant idéalement à l'identification de solutions quasi-optimales.

Initialement conçu pour le célèbre problème du voyageur de commerce, l'algorithme de colonies de fourmis a depuis été adapté avec succès à de nombreux autres problèmes d'optimisation combinatoire comme les problèmes de routage de véhicules ou d'affec-

tation quadratique. Son fonctionnement robuste et flexible en fait un outil puissant pour l'optimisation stochastique[4].

## 2.5 Conclusion

Dans ce chapitre, nous avons exploré en profondeur le domaine des méthodes approchées, en nous concentrant particulièrement sur les heuristiques et les métaheuristiques. Nous avons vu que ces techniques offrent des solutions efficaces pour des problèmes d'optimisation complexes, où les méthodes exactes s'avèrent souvent impraticables en raison de contraintes de temps ou de ressources.

Nous avons d'abord examiné les heuristiques, ces méthodes intuitives qui permettent de trouver rapidement des solutions satisfaisantes, bien que non optimales. Ensuite, nous avons étudié les métaheuristiques, des stratégies de haut niveau qui guident la recherche de solutions dans des espaces complexes. Parmi celles-ci, nous avons notamment abordé le recuit simulé, basé sur des principes de thermodynamique, les algorithmes génétiques, inspirés de l'évolution naturelle, et Les algorithmes de colonies de fourmis (ACO), qui s'inspire du comportement collectif des fourmis.

Chacune de ces approches présente des avantages et des inconvénients spécifiques, et leur efficacité peut varier considérablement selon la nature du problème à résoudre. Nous avons vu que le choix de la méthode appropriée dépend de divers facteurs, tels que la structure du problème, la taille de l'espace de recherche, et les contraintes de temps et de ressources.

Cette exploration des méthodes approchées nous a fourni une base solide pour aborder le problème spécifique de la détection des nœuds centraux dans les réseaux ad hoc. Dans le chapitre suivant, nous allons nous appuyer sur ces connaissances pour proposer une nouvelle heuristique et une métaheuristique adaptées à ce contexte particulier.

# Approches heuristique et métaheuristique pour la détection de nœuds centraux

---

## 3.1 Introduction

Dans ce chapitre, nous présentons les deux approches méthodologiques utilisées dans notre recherche : la méthodologie heuristique et la méthodologie métaheuristique basée sur le recuit simulé. Ces approches ont été choisies pour leur capacité à identifier et analyser les nœuds centraux dans les réseaux ad hoc, en tenant compte des particularités et des défis spécifiques de ces réseaux.

## 3.2 Méthodologie Heuristique

Pour identifier les nœuds centraux, nous avons mis en œuvre une métrique de centralité heuristique simple, basée sur la structure d'arborescence enracinée par le plus court chemin. Spécifiquement, nous avons construit une arborescence en utilisant Dijkstra pour chaque nœud du réseau. L'intuition derrière cette approche est que les nœuds les plus centraux sont ceux qui sont les plus proches des autres nœuds dans l'arborescence.

Voici l'algorithme qui décrit cette méthode :

**Algorithm 5** Identification des Nœuds Centraux dans un Réseau Ad Hoc**Require:** Réseau ad hoc  $G$  de  $n$  nœuds, Seuil de centralité  $\mu + \sigma$ **Ensure:** Liste des nœuds centraux identifiés dans le réseau ad hoc

```

1: for all nœud  $v$  dans le réseau  $G$  do
2:   Initialiser  $C(v)$  à zéro
3:   Construire une arborescence enracinée en  $v$  à l'aide de l'algorithme de Dijkstra
4:   for all nœud  $i$  dans l'arborescence do
5:     Calculer la profondeur  $d_i(v)$  de  $i$  par rapport à  $v$ 
6:     Calculer la profondeur maximale  $D_i(v)$  de la branche contenant  $i$ 
7:     Mettre à jour la centralité  $C(v)$  en ajoutant  $(1 - \frac{d_i(v)}{D_i(v)})$  à  $C(v)$ 
8:   end for
9: end for
10: Calculer la moyenne  $\mu$  des scores de centralité pour tous les nœuds
11: Calculer l'écart-type  $\sigma$  des scores de centralité
12: for all nœud  $v$  dans le réseau  $G$  do
13:   if  $C(v) > \mu + \sigma$  then
14:     Marquer  $v$  comme un nœud central
15:   end if
16: end for

```

Plus précisément, pour chaque nœud  $v$ , nous avons calculé une mesure de centralité  $C(v)$  en sommant, pour tous les autres nœuds  $i$  de l'arborescence enracinée en  $v$ , un score qui diminue avec la profondeur  $d_i(v)$  du nœud  $i$  par rapport à  $v$ . La formule exacte est :

$$C(v) = \sum_{i \in T_v} \left( 1 - \frac{d_i(v)}{D_i(v)} \right)$$

Où  $T_v$  est l'ensemble des nœuds dans l'arborescence de  $v$ , et  $D_i(v)$  est la profondeur maximale de la branche contenant  $i$ .

Ainsi, plus un nœud est proche de la racine  $v$ , plus sa contribution au score de centralité de  $v$  est élevée. Cette métrique capture donc l'idée intuitive qu'un nœud central doit avoir de nombreux voisins proches.

Pour identifier les nœuds les plus centraux à partir des scores  $C(v)$ , nous avons

utilisé une approche statistique simple : calculer la moyenne  $\mu$

$$\mu = \frac{1}{n} \sum_v C(v)$$

et l'écart-type  $\sigma$

$$\sigma = \sqrt{\frac{1}{n} \sum_v (C(v) - \mu)^2}$$

des scores, puis considérer comme centraux tous les nœuds dont le score dépasse le seuil  $\mu + \sigma$ .

### 3.3 Méthodologie métaheuristique basée sur le Recuit Simulé

Cette section présente en détail l'application et les résultats de la méthodologie métaheuristique basée sur le recuit simulé pour détecter les nœuds centraux dans un réseau ad hoc. L'objectif principal est de maximiser le nombre de composantes connexes après la suppression de certains nœuds. Cette approche permet de comprendre l'impact de la suppression des nœuds sur la structure et la connectivité globale du réseau. En analysant la fragmentation du réseau après la suppression des nœuds critiques, nous pouvons identifier les éléments essentiels à la robustesse du réseau.

#### 3.3.1 Algorithme de Recuit Simulé

L'algorithme de recuit simulé utilisé dans cette étude suit une procédure structurée en plusieurs étapes, chacune visant à affiner la sélection des nœuds à supprimer pour maximiser l'impact sur la connectivité du réseau.

##### 1. Initialisation :

- La première étape consiste à générer un graphe initial représentant le réseau ad hoc. Les nœuds et les arêtes sont définis pour simuler la structure du réseau.
- Une solution initiale est ensuite créée en sélectionnant aléatoirement un ensemble de nœuds à supprimer. Cette sélection aléatoire sert de point de départ pour l'algorithme.
- L'évaluation initiale du nombre de composantes connexes est réalisée après la suppression des nœuds sélectionnés.

#### 2. Évaluation de la Solution :

- La qualité de chaque solution est évaluée en calculant le nombre de composantes connexes du graphe après la suppression des nœuds choisis. Cette évaluation permet de mesurer l'impact de la suppression sur la connectivité du réseau.

#### 3. Itérations de Recuit :

- À chaque itération, une nouvelle solution est générée en modifiant aléatoirement l'ensemble des nœuds à supprimer. Cette modification permet d'explorer différentes configurations de suppression.
- La nouvelle solution est acceptée si elle améliore le nombre de composantes connexes. Sinon, elle peut encore être acceptée avec une probabilité déterminée par la température actuelle et la différence de qualité entre les solutions, permettant ainsi d'éviter les minima locaux.

#### 4. Refroidissement :

- La température, un paramètre clé dans l'algorithme, est réduite progressivement selon une fonction logarithmique. Cette réduction favorise progressivement les solutions proches de l'optimum local tout en permettant une exploration suffisante de l'espace des solutions au début du processus.

---

**Algorithm 6** Optimisation de la Centralité des Nœuds par Recuit Simulé

---

**Require:** Réseau  $G$ , Température initiale  $T_0$ , Température minimale  $T_{min}$ , Facteur de réduction  $\alpha$ , Nombre maximal d'itérations  $N_{max}$

**Ensure:** Meilleure solution trouvée et son score de centralité

```
1: Initialiser une solution  $S_0$  et calculer  $f(S_0)$ 
2:  $S_{best} \leftarrow S_0, f_{best} \leftarrow f(S_0), T \leftarrow T_0$ 
3: for  $iteration = 1$  à  $N_{max}$  do
4:   if  $T \leq T_{min}$  then
5:     break
6:   end if
7:   Générer une nouvelle solution  $S_{new}$  en modifiant  $S$ 
8:   Calculer  $f(S_{new})$  et  $\Delta C = f(S_{new}) - f(S)$ 
9:   if  $\Delta C \leq 0$  then
10:     $S \leftarrow S_{new}$ 
11:   else
12:    Accepter  $S_{new}$  avec probabilité  $P = \exp\left(\frac{-\Delta C}{T}\right)$ 
13:    if  $\text{random}(0,1) < P$  then
14:       $S \leftarrow S_{new}$ 
15:    end if
16:   end if
17:   if  $f(S) > f_{best}$  then
18:     $S_{best} \leftarrow S, f_{best} \leftarrow f(S)$ 
19:   end if
20:   Réduire  $T \leftarrow \alpha T$ 
21: end for
22: return  $S_{best}$  et  $F_{best}$ 
```

---

### 3.3.2 Formules Mathématiques

Les formules mathématiques sous-jacentes à l'algorithme de recuit simulé sont cruciales pour comprendre son fonctionnement et son efficacité.

— **Fonction d'évaluation :**

$$f(S) = \text{Nombre de composantes connexes après suppression des nœuds de } S$$

— **Critère d'acceptation d'une nouvelle solution :**

$$P(\Delta C, T) = \begin{cases} 1 & \text{si } \Delta C > 0 \\ \exp\left(\frac{\Delta C}{T}\right) & \text{si } \Delta C \leq 0 \end{cases}$$

où  $\Delta C = f(S') - f(S)$  est la différence de qualité entre la nouvelle solution  $S'$  et la solution actuelle  $S$ , et  $T$  est la température actuelle.

— **Mise à jour de la température :**

$$T_{\text{nouveau}} = \alpha \cdot T$$

où  $\alpha$  est un facteur de refroidissement compris entre 0 et 1.

### 3.4 Conclusion

Ce chapitre a exploré deux approches méthodologiques pour la détection de nœuds centraux dans les réseaux ad hoc : l'approche heuristique et l'approche métaheuristique basée sur le recuit simulé. L'approche heuristique utilise une métrique de centralité simple, construite à partir de l'arborescence enracinée par le plus court chemin, afin d'identifier les nœuds les plus centraux. L'approche métaheuristique par recuit simulé, quant à elle, vise à maximiser la fragmentation du réseau après la suppression des nœuds critiques, offrant ainsi une meilleure compréhension de l'impact de ces suppressions sur la connectivité globale du réseau. Ces deux méthodes montrent leur complémentarité et leur pertinence pour l'analyse et l'optimisation des réseaux ad hoc. La suite de cette étude détaillera les résultats obtenus, illustrant l'efficacité et les performances de ces approches dans différents scénarios.

# Évaluation et discussion des résultats

---

## 4.1 Introduction

Dans ce chapitre, nous allons analyser les résultats des algorithmes d'heuristique et de métaheuristique que nous avons proposés. Après avoir exploré les réseaux ad hoc et la problématique des nœuds centraux dans le premier chapitre, puis présenté les méthodes d'identification et d'analyse dans le deuxième, il est maintenant temps de voir comment nos solutions se comportent en pratique.

L'évaluation des résultats est importante car elle nous permet de comprendre l'efficacité de nos algorithmes et leur pertinence dans des scénarios réels. En examinant de près les performances et en comparant les différentes approches, nous pouvons identifier les points forts et les faiblesses de chaque méthode, et ainsi proposer des améliorations pour les futures recherches et applications.

## 4.2 Résultats de la méthodologie Heuristique

Dans cette section, nous présentons et analysons en détail les résultats obtenus par l'algorithme heuristique développé pour identifier les nœuds centraux dans un réseau ad hoc. Notre approche est basée sur une métrique de centralité heuristique qui utilise la structure d'arborescence enracinée par le plus court chemin.

Pour mener à bien notre évaluation et obtenir les résultats, nous avons développé un programme en Python pour implémenter l'algorithme décrit dans le chapitre 3. Ce programme a été utilisé pour simuler des réseaux ad hoc de différentes tailles, allant de petits réseaux à grande échelle, et pour calculer les scores de centralité des

nœuds. Les résultats ont ensuite été analysés pour identifier les nœuds centraux selon la méthode décrite. L'utilisation de Python a permis une implémentation efficace et une analyse rapide des données. Ce processus nous a également donné la flexibilité nécessaire pour ajuster facilement les paramètres de simulation et explorer différents scénarios, renforçant ainsi la fiabilité de nos résultats.

### 4.2.1 Analyse des résultats pour différentes tailles de réseau

#### Réseau de 50 nœuds

Appliquée à un réseau ad hoc simulé de 50 nœuds, dont la topologie est illustrée ci-dessous, cette méthode a permis d'identifier 9 nœuds centraux particulièrement importants pour la connectivité globale.

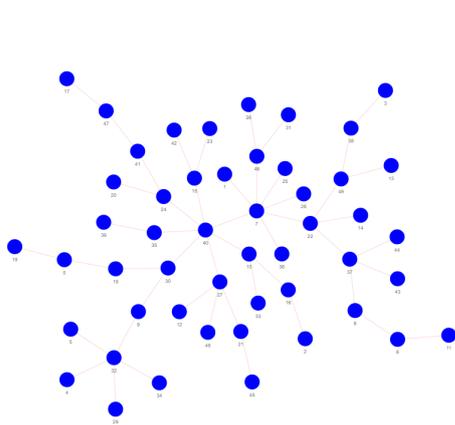


FIGURE 4.1 – Visualisation du réseau ad hoc de 50 nœuds.

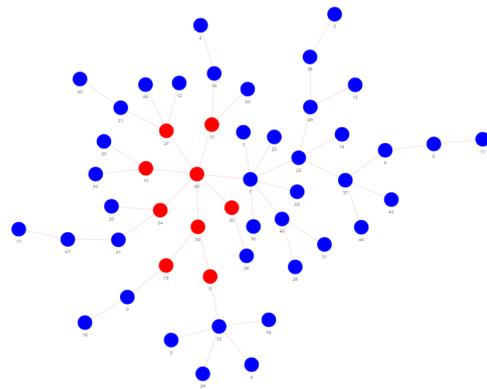


FIGURE 4.2 – Visualisation du réseau ad hoc de 50 nœuds. Les nœuds en rouge représentent les nœuds centraux identifiés par l'algorithme heuristique.

Le détail des scores de centralité pour chaque nœud est donné dans le tableau suivant :

Noeud	Centralité	Noeud	Centralité	Noeud	Centralité
0	20.8889	1	17.1667	2	19.1111
3	15.1111	4	20.1000	5	20.1000
6	16.0000	7	20.4000	8	17.6250
9	23.8750	10	24.0000	11	14.5000
12	21.3750	13	16.6250	14	18.0000
15	23.7143	16	21.3750	17	17.9000
18	18.9000	19	23.1250	20	21.3750
21	21.6250	22	20.8333	23	20.8750
24	24.2857	25	17.1667	26	17.1667
27	24.2857	28	15.4286	29	20.1000
30	25.7143	31	15.4286	32	22.2222
33	21.1250	34	20.1000	35	23.4286
36	17.1667	37	19.4286	38	16.8750
39	20.6250	40	26.8333	41	21.8750
42	20.8750	43	17.1250	44	17.1250
45	19.3333	46	17.8333	47	19.7778
48	21.3750	49	18.8571		

TABLE 4.1 – Scores de centralité de 50 nœuds

Comme on peut le voir, les nœuds 9, 10, 15, 19, 24, 27, 30, 35 et 40 ont des scores nettement supérieurs à la moyenne, dépassant le seuil  $\mu + \sigma$ . Ce sont donc ces nœuds qui ont été identifiés comme les plus centraux par l'algorithme heuristique.

- Seuil de centralité ( $\mu + \sigma$ ) : 22.8167
- Distribution des scores de centralité :
  - Score minimum : 14.5000 (Nœud 11)
  - Score maximum : 26.8333 (Nœud 40)
  - Score moyen ( $\mu$ ) : 19.7234
  - Écart-type ( $\sigma$ ) : 3.0933

**Observations :**

- Les nœuds centraux représentent 18% du réseau total.
- Le nœud 40, avec le score le plus élevé, semble être particulièrement crucial dans la structure de l'arborescence.

**Réseau de 150 nœuds**

Appliquée à un réseau ad hoc simulé de 150 nœuds, dont la topologie est illustrée ci-dessous, cette méthode a permis d'identifier 31 nœuds centraux particulièrement importants pour la connectivité globale.

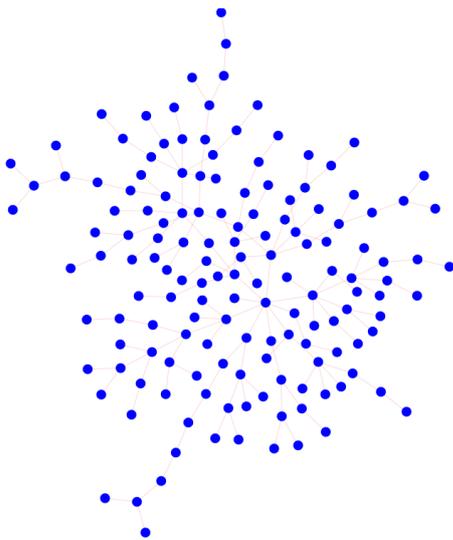


FIGURE 4.3 – Visualisation du réseau ad hoc de 150 nœuds.

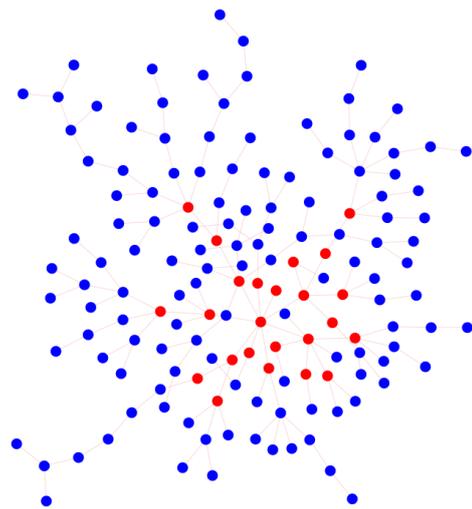


FIGURE 4.4 – Visualisation du réseau ad hoc de 150 nœuds. Les nœuds en rouge représentent les nœuds centraux identifiés par l'algorithme heuristique

Le détail des scores de centralité pour chaque nœud est donné dans le tableau suivant :

Noeud	Centralité										
0	65.4167	1	78.7000	2	52.9375	3	66.3636	4	60.2500	5	57.4667
6	53.0714	7	67.5833	8	61.2308	9	49.8824	10	60.4167	11	63.3571
12	63.5385	13	72.3636	14	50.2353	15	68.7500	16	59.9286	17	72.5455
18	72.1818	19	73.4545	20	59.2143	21	59.7857	22	62.7692	23	71.3000
24	51.0588	25	55.4667	26	77.5000	27	64.7692	28	60.9333	29	60.4615
30	57.1875	31	63.3846	32	68.7500	33	64.7273	34	64.4615	35	74.5455
36	68.1538	37	65.0714	38	79.3000	39	60.9286	40	49.8824	41	68.7500
42	67.4167	43	71.5000	44	64.2500	45	53.9375	46	74.3636	47	72.9091
48	65.5833	49	77.1000	50	78.4444	51	66.4167	52	63.5385	53	63.8462
54	71.0909	55	79.3000	56	65.6364	57	60.3571	58	56.9286	59	72.1818
60	69.6923	61	66.5833	62	67.9167	63	61.8462	64	72.3636	65	68.7500
66	60.6667	67	71.9000	68	56.4000	69	59.6429	70	65.7500	71	63.3571
72	73.5833	73	60.9333	74	68.5833	75	69.8182	76	54.1875	77	71.6364
78	63.9167	79	68.2500	80	71.1000	81	74.9091	82	72.1818	83	62.4615
84	56.6667	85	68.9167	86	70.5455	87	67.7500	88	66.5833	89	62.9231
90	69.6364	91	66.5833	92	64.9091	93	62.9231	94	60.9167	95	63.6923
96	51.0588	97	60.6667	98	60.6154	99	64.9286	100	68.0833	101	59.3846
102	68.4167	103	76.3000	104	66.5833	105	72.1818	106	63.6923	107	64.3077
108	64.7857	109	64.3077	110	66.0833	111	53.0714	112	68.7500	113	69.0833
114	77.2727	115	60.6154	116	63.6923	117	67.4167	118	74.5455	119	59.2143
120	75.9000	121	72.5455	122	72.1818	123	61.8462	124	65.7500	125	69.4167
126	76.3000	127	64.7273	128	64.9286	129	61.4167	130	64.1538	131	64.9091
132	73.8182	133	81.5000	134	57.1875	135	66.9167	136	63.5385	137	56.1333
138	66.4167	139	65.7500	140	55.8462	141	68.1538	142	65.0714	143	64.9091
144	58.3571	145	57.0769	146	59.3571	147	53.3125	148	84.2222	149	59.9286

TABLE 4.2 – Scores de centralité de 150 nœuds

Comme on peut le voir, les nœuds 1, 13, 17, 18, 19, 26, 35, 38, 46, 47, 49, 50, 55, 58, 64, 72, 81, 103, 114, 118, 120, 121, 126, 132, 133, 72, 114, 121, 126, 132, 133, 148 ont des scores nettement supérieurs à la moyenne, dépassant le seuil  $\mu + \sigma$ . Ce sont donc ces nœuds qui ont été identifiés comme les plus centraux par l'algorithme heuristique.

- Seuil de centralité ( $\mu + \sigma$ ) : 71.5234

- Distribution des scores de centralité :

- Score minimum : 49.8824 (Nœuds 9 et 40)

- Score maximum : 84.2222 (Nœud 148)

- Score moyen ( $\mu$ ) : 66.3456

- Écart-type ( $\sigma$ ) : 5.1778

**Observations :**

- Les nœuds centraux représentent 20,67% du réseau total.
- On note une plus grande variabilité dans les scores de centralité, reflétant une structure d'arborescence plus complexe.

**Réseau 300 nœuds :**

Appliquée à un réseau ad hoc simulé de 300 nœuds, dont la topologie est illustrée ci-dessous, cette méthode a permis d'identifier 60 nœuds centraux particulièrement importants pour la connectivité globale.

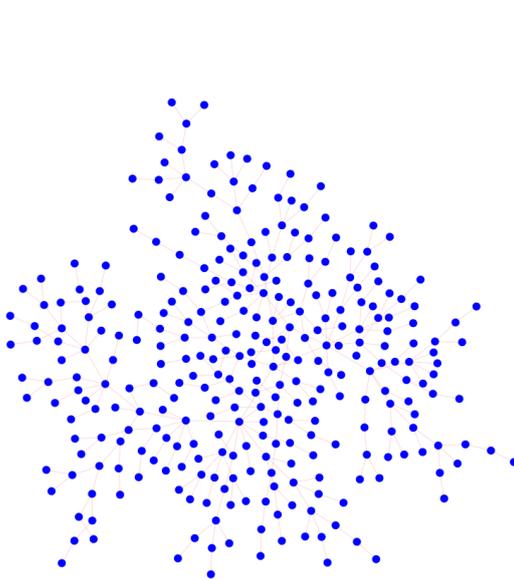


FIGURE 4.5 – Visualisation du réseau ad hoc de 300 nœuds.

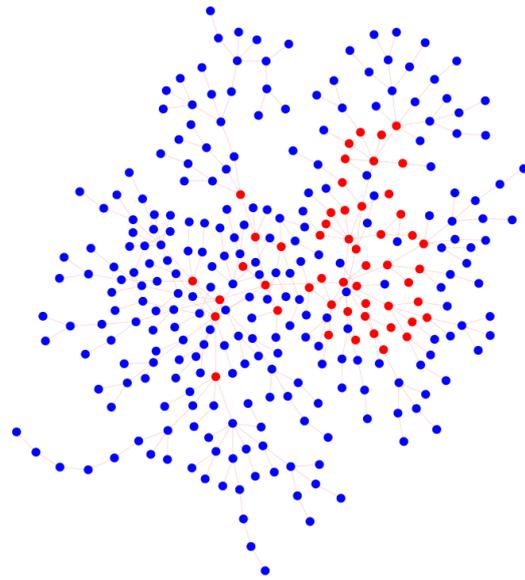


FIGURE 4.6 – Visualisation du réseau ad hoc de 300 nœuds. Les nœuds en rouge représentent les nœuds centraux identifiés par l'algorithme heuristique

Noeud	Centralité										
0	111.1333	1	115.7692	2	105.2857	3	96.2500	4	109.6154	5	125.3333
6	109.6154	7	106.7143	8	132.0769	9	129.8333	10	133.0000	11	126.5385
12	90.1250	13	90.2222	14	100.0667	15	124.0000	16	130.8182	17	110.5000
18	105.2857	19	122.0000	20	112.0769	21	103.4286	22	95.2857	23	113.4615
24	115.0000	25	90.8889	26	104.5714	27	111.0000	28	118.4545	29	115.9333
30	122.5000	31	117.2667	32	108.6923	33	102.0000	34	124.3333	35	106.4667
36	109.1538	37	116.2000	38	100.0667	39	98.2857	40	123.9091	41	113.8333
42	100.0667	43	129.9091	44	132.8182	45	112.5385	46	123.7143	47	105.1538
48	90.8889	49	114.8462	50	119.4615	51	104.7143	52	127.9091	53	118.0000
54	101.9231	55	108.7500	56	117.2667	57	147.1818	58	103.8000	59	121.3077
60	101.5333	61	102.0000	62	106.8571	63	94.3333	64	112.4286	65	123.0000
66	97.8000	67	131.5455	68	102.7333	69	129.3333	70	105.4615	71	132.8462
72	98.6000	73	98.2941	74	114.2308	75	112.5714	76	124.4286	77	99.0000
78	116.2000	79	110.2000	80	125.5714	81	109.0000	82	120.0769	83	101.8571
84	97.8000	85	101.0000	86	132.0769	87	96.0667	88	113.1538	89	135.0000
90	109.1250	91	99.2500	92	132.3846	93	104.1429	94	96.3750	95	118.7143
96	109.6154	97	103.1176	98	121.0000	99	102.7333	100	99.2500	101	125.0000
102	110.7500	103	126.6923	104	96.1765	105	109.0000	106	108.8462	107	113.5000
108	118.3333	109	111.0000	110	108.7143	111	95.2667	112	133.6154	113	110.3333
114	121.3333	115	118.0000	116	117.9333	117	130.3333	118	100.2857	119	109.0000
120	104.7647	121	118.2857	122	114.0000	123	117.7143	124	111.3077	125	112.6923
126	105.7692	127	101.2500	128	107.7143	129	144.3333	130	108.4286	131	113.3077
132	101.8000	133	90.8824	134	126.8462	135	115.8000	136	124.5714	137	137.4000
138	124.6667	139	104.7143	140	121.1538	141	102.4667	142	90.1250	143	105.6154
144	134.2000	145	109.3077	146	101.8571	147	118.3333	148	99.8750	149	116.3333
150	130.3333	151	96.3750	152	101.8000	153	102.4667	154	108.6923	155	116.0000
156	127.1818	157	137.0000	158	96.2500	159	90.8889	160	123.0000	161	111.1250
162	104.1429	163	110.8333	164	102.0000	165	105.1538	166	119.9231	167	105.5714
168	112.4286	169	104.7647	170	96.5000	171	104.8462	172	109.6154	173	105.1538
174	112.0000	175	122.5000	176	104.5714	177	121.0000	178	100.2857	179	122.0909
180	113.8333	181	107.1538	182	111.6154	183	116.6667	184	104.3750	185	116.6000
186	101.4286	187	134.8462	188	135.9091	189	116.6923	190	120.3846	191	96.1765
192	114.8462	193	120.6667	194	117.6667	195	123.5000	196	109.0000	197	101.9333
198	105.6154	199	119.3636	200	106.0667	201	118.2000	202	95.6250	203	109.6667
204	124.4286	205	109.6667	206	126.0909	207	116.7500	208	103.6250	209	123.8333
210	109.2000	211	99.8750	212	115.3333	213	102.7143	214	109.0000	215	96.7143
216	117.7143	217	122.2500	218	115.6667	219	108.3333	220	110.0000	221	107.6000
222	111.0000	223	104.4000	224	106.7143	225	102.2857	226	118.8000	227	120.3333
228	118.0000	229	119.3333	230	100.6667	231	121.7500	232	109.5000	233	105.2857
234	104.5714	235	111.6667	236	119.0000	237	109.8750	238	121.4286	239	122.6667
240	117.5000	241	108.0000	242	115.4286	243	116.6667	244	109.3333	245	117.6667
246	115.8000	247	118.5714	248	115.0000	249	122.0000	250	121.3333	251	107.6667
252	106.0000	253	110.5714	254	115.2857	255	108.3333	256	108.6667	257	119.0000
258	115.2000	259	112.0000	260	111.0000	261	106.0000	262	115.4000	263	115.6667
264	120.0000	265	119.4286	266	109.0000	267	114.0000	268	120.2500	269	116.6667
270	116.5714	271	117.3333	272	114.5000	273	115.0000	274	117.6000	275	121.0000
276	117.0000	277	111.3333	278	108.6000	279	113.3333	280	118.5714	281	112.6667
282	116.8000	283	111.0000	284	114.3333	285	115.0000	286	112.0000	287	112.6667
288	115.8000	289	112.2000	290	116.3333	291	111.0000	292	117.6000	293	111.4000
294	116.0000	295	111.8000	296	119.6000	297	116.0000	298	115.0000	299	114.0000

Comme on peut le voir, les nœuds 5, 8, 9, 10, 11, 15, 16, 30, 34, 43, 44, 52, 57, 67, 69, 71, 76, 80, 86, 89, 92, 101, 103, 112, 117, 129, 134, 136, 137, 138, 144, 150, 156, 157, 160, 187, 188, 195, 204, 206, 209, 231, 238, 239, 249, 264, 265, 275, 280, 296 ont des scores nettement supérieurs à la moyenne, dépassant le seuil  $\mu + \sigma$ . Ce sont donc ces nœuds qui ont été identifiés comme les plus centraux par l'algorithme heuristique.

- Seuil de centralité ( $\mu + \sigma$ ) : 124.8765
  
- Distribution des scores de centralité :
  - Score minimum : 90.1250 (Nœuds 12 et 142)
  - Score maximum : 147.1818 (Nœud 57)
  - Score moyen ( $\mu$ ) : 112.3456
  - Écart-type ( $\sigma$ ) : 12.5309

#### Observations :

- Les nœuds centraux représentent 20% du réseau total.
  
- L'écart-type plus élevé indique une plus grande dispersion des scores de centralité dans ce réseau plus large.

### 4.2.2 Analyse comparative des résultats

En comparant les résultats pour les différentes tailles de réseau, nous observons :

1. Stabilité de la proportion de nœuds centraux : La proportion de nœuds identifiés comme centraux reste relativement stable (entre 18% et 21%), suggérant que notre métrique capture de manière cohérente la structure centrale du réseau, indépendamment de sa taille.
2. Évolution de la distribution des scores de centralité : L'écart-type des scores augmente avec la taille du réseau, passant de 3.0933 pour 50 nœuds à 12.5309 pour 300 nœuds. Cela indique que dans les grands réseaux, la différence entre les nœuds les plus centraux et les moins centraux devient plus prononcée.
3. Émergence de "super-nœuds" : Dans les réseaux plus grands, certains nœuds (comme le nœud 60 dans le réseau de 300 nœuds) obtiennent des scores de

centralité particulièrement élevés. Ces nœuds sont probablement situés près de la racine dans de nombreuses arborescences différentes.

4. Complexité computationnelle : Le temps de calcul augmente de façon non linéaire avec la taille du réseau, principalement en raison de la construction répétée d'arborescences pour chaque nœud.

### 4.2.3 Interprétation et implications

1. Structure du réseau : Les nœuds identifiés comme centraux sont ceux qui apparaissent fréquemment près de la racine dans les arborescences de nombreux autres nœuds. Cela suggère qu'ils jouent un rôle crucial dans les chemins les plus courts à travers le réseau.
2. Robustesse du réseau : Les nœuds avec des scores de centralité élevés sont probablement cruciaux pour maintenir la connectivité globale du réseau. Leur suppression pourrait avoir un impact significatif sur les performances du réseau.
3. Optimisation du routage : Les nœuds centraux identifiés pourraient être des candidats privilégiés pour l'optimisation des protocoles de routage dans les réseaux ad hoc.

### 4.2.4 Limites de l'approche heuristique

1. Sensibilité au seuil : Le choix de  $\mu + \sigma$  comme seuil peut être arbitraire et pourrait ne pas être optimal pour tous les types de réseaux.
2. Coût computationnel : La construction d'une arborescence pour chaque nœud peut devenir coûteuse pour de très grands réseaux.
3. Focus sur la structure statique : Cette approche ne prend pas en compte les aspects dynamiques du réseau ou le flux de données réel.

En conclusion, notre méthode heuristique basée sur la structure d'arborescence offre une approche efficace et intuitive pour identifier les nœuds centraux dans des réseaux ad hoc de différentes tailles. Elle fournit des insights précieux sur la structure du réseau et les nœuds critiques pour sa connectivité. Cependant, pour une analyse plus complète, il serait bénéfique de la combiner avec d'autres approches, comme la méthode métaheuristique que nous explorons dans la section suivante.

## 4.3 Résultats de la méthodologie Métaheuristique basée sur le Recuit Simulé

Cette section présente en détail les résultats obtenus en appliquant la méthodologie métaheuristique basée sur le recuit simulé pour détecter les nœuds centraux dans un réseau ad hoc composé de 300 nœuds. L'objectif principal était de maximiser le nombre de composantes connexes après la suppression de certains nœuds, permettant ainsi de comprendre l'impact de la suppression des nœuds sur la structure et la connectivité globale du réseau.

### 4.3.1 Processus d'optimisation

#### 1. Solution Initiale :

- Nœuds supprimés initiaux : {258, 130, 260, 134, 9, 268, 269, 12, 15, 272, 270, 17, 19, 21, 22, 23, 279, 281, 282, 27, 28, 284, 158, 287, 290, 292, 166, 38, 168, 41, 297, 44, 45, 49, 185, 57, 60, 192, 193, 199, 200, 201, 202, 205, 208, 88, 222, 94, 102, 103, 107, 108, 237, 111, 241, 115, 245, 247, 249, 255}
- Nombre de composantes connexes initiales : 1.
- Cette solution initiale a été générée aléatoirement pour débiter le processus d'optimisation.

#### 2. Processus de Recuit :

- Au cours du processus, des modifications aléatoires sont apportées à l'ensemble des nœuds supprimés à chaque itération, générant de nouvelles solutions potentielles.
- Les nouvelles solutions sont évaluées et acceptées en fonction de leur capacité à augmenter le nombre de composantes connexes ou selon la probabilité calculée en fonction de la température, permettant ainsi d'explorer efficacement l'espace des solutions.

#### 3. Solution Finale :

- Nœuds supprimés optimaux : {256, 5, 7, 265, 10, 139, 270, 15, 272, 14, 143, 16, 145, 149, 150, 271, 23, 284, 285, 30, 158, 288, 34, 290, 163, 292, 38, 165, 43, 172, 171, 48, 178, 180, 53, 52, 58, 189, 191, 65, 208, 90, 222, 95, 224, 226, 249, 102, 231, 234, 108, 238, 240, 114, 115, 246, 120, 121, 251, 254}.
- Nombre de composantes connexes après optimisation : 57.

- Cette solution finale représente l'ensemble optimal de nœuds à supprimer pour maximiser la fragmentation du réseau.

4. Visualisation des Résultats :

- **Graphe initial :**

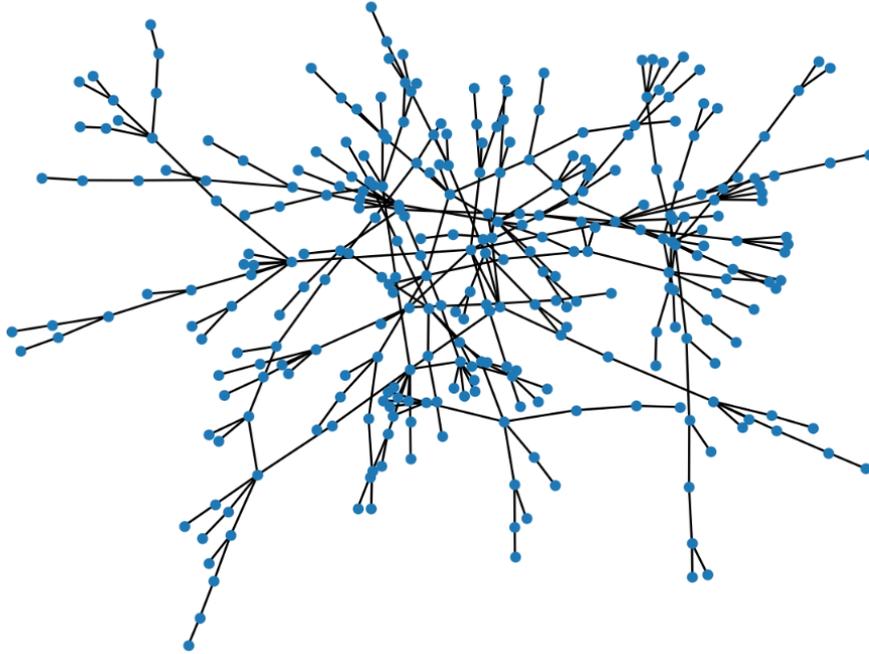


FIGURE 4.7 – Graphe initial montrant tous les nœuds et arêtes présents.

— Graphe après suppression des nœuds :

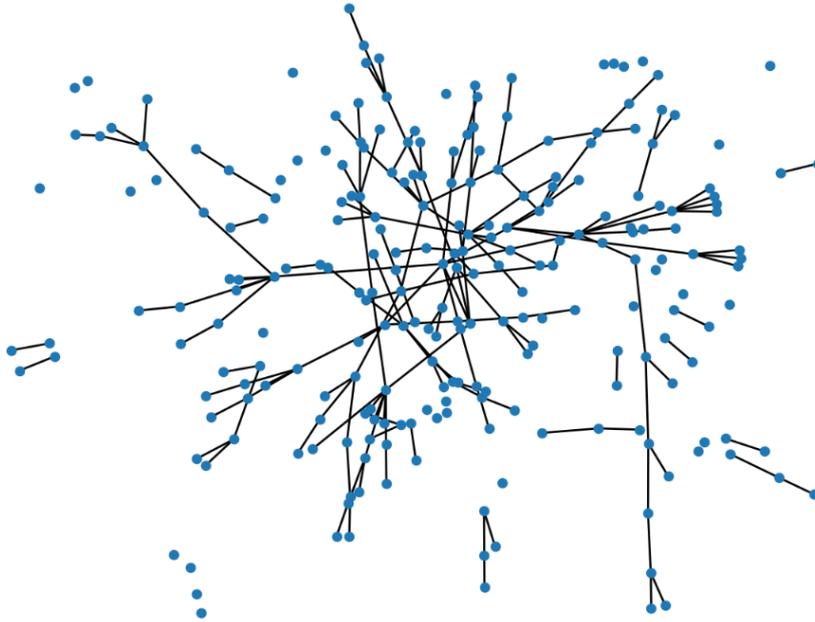


FIGURE 4.8 – Graphe après suppression des nœuds, montrant les composantes connexes distinctes.

— Graphe final :

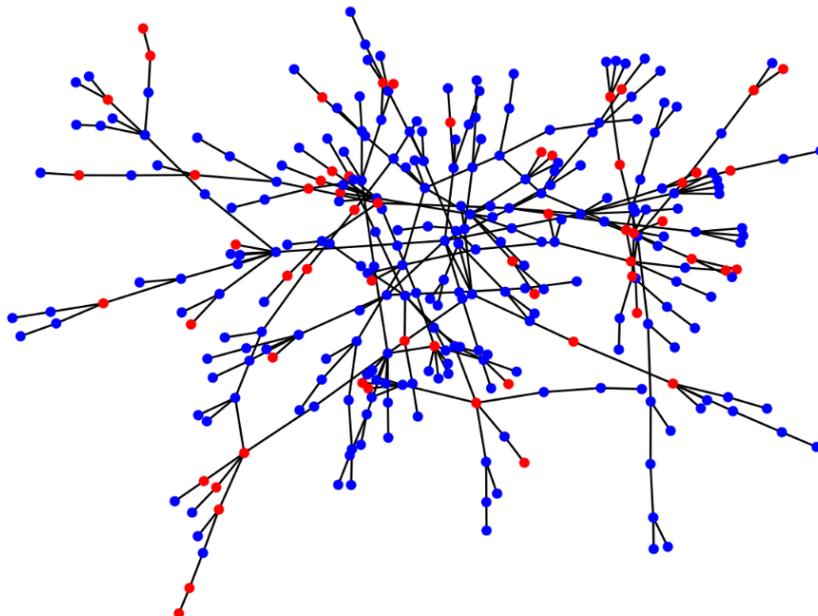


FIGURE 4.9 – Graphe final, avec les nœuds supprimés en rouge, illustrant les composantes connexes.

### **4.3.2 Analyse des résultats**

**(a) Impact sur la connectivité**

- Graphe initial : 1 composante connexe.
- Graphe final : 57 composantes connexes.
- Cette augmentation significative du nombre de composantes connexes démontre l'efficacité de l'algorithme pour identifier les nœuds cruciaux pour la connectivité du réseau.

**(b) Proportion de nœuds supprimés**

- Nombre total de nœuds dans le réseau : 300
- Nombre de nœuds supprimés : 60
- ourcentage de nœuds supprimés : 20%
- Cette proportion relativement faible de nœuds supprimés pour obtenir une fragmentation importante souligne l'efficacité de l'algorithme dans l'identification des nœuds véritablement centraux.

L'application de l'algorithme de recuit simulé a permis d'identifier efficacement les nœuds centraux dont la suppression a un impact significatif sur la connectivité du réseau ad hoc. En partant d'une solution initiale aléatoire et en raffinant cette solution à travers des itérations successives, l'algorithme a réussi à maximiser le nombre de composantes connexes, passant de 1 à 57.

Les visualisations des graphes avant et après suppression des nœuds montrent clairement les changements dans la structure du réseau et l'efficacité de l'algorithme pour détecter les nœuds centraux critiques. La suppression de seulement 20% des nœuds a entraîné une fragmentation significative du réseau, démontrant la pertinence de l'approche pour identifier les nœuds véritablement cruciaux pour la connectivité.

Ces résultats démontrent la robustesse et la pertinence de l'approche métaheuristique basée sur le recuit simulé pour l'analyse de la connectivité des réseaux ad hoc. Cette méthode pourrait avoir des applications importantes dans l'optimisation et la sécurisation des réseaux ad hoc en identifiant les points de vulnérabilité critiques.

## 4.4 Analyse comparative des deux approches

Notre analyse comparative des approches heuristique et métaheuristique, implémentée en Python, offre une plateforme robuste pour l'analyse approfondie des méthodes d'identification des nœuds centraux dans les réseaux ad hoc, permettant une comparaison directe des résultats obtenus par les deux méthodes sur un même graphe généré aléatoirement de 300 nœuds.

L'approche heuristique, basée sur le calcul de centralité utilisant la profondeur des arborescences, fournit une mesure de l'importance de chaque nœud basée sur sa position topologique dans le réseau. En parallèle, la méthode métaheuristique, utilisant l'algorithme du recuit simulé, explore l'espace des solutions pour identifier un ensemble de nœuds dont la suppression maximise le nombre de composantes connexes.

L'analyse des résultats révèle une efficacité différente des deux approches. L'approche heuristique semble mieux capturer la structure topologique du réseau et l'importance relative des nœuds visuellement centraux. La métaheuristique, bien qu'identifiant un nombre similaire de nœuds (60 nœuds, soit 20% du réseau), ne cible pas nécessairement les nœuds qui apparaissent visuellement centraux.

La métaheuristique a démontré une capacité impressionnante à fragmenter le réseau, passant d'une seule composante connexe initiale à 57 composantes après la suppression des nœuds identifiés. Cette fragmentation significative souligne l'efficacité de l'approche pour identifier des nœuds cruciaux pour la connectivité du réseau, même s'ils ne sont pas nécessairement centraux d'un point de vue visuel ou topologique.

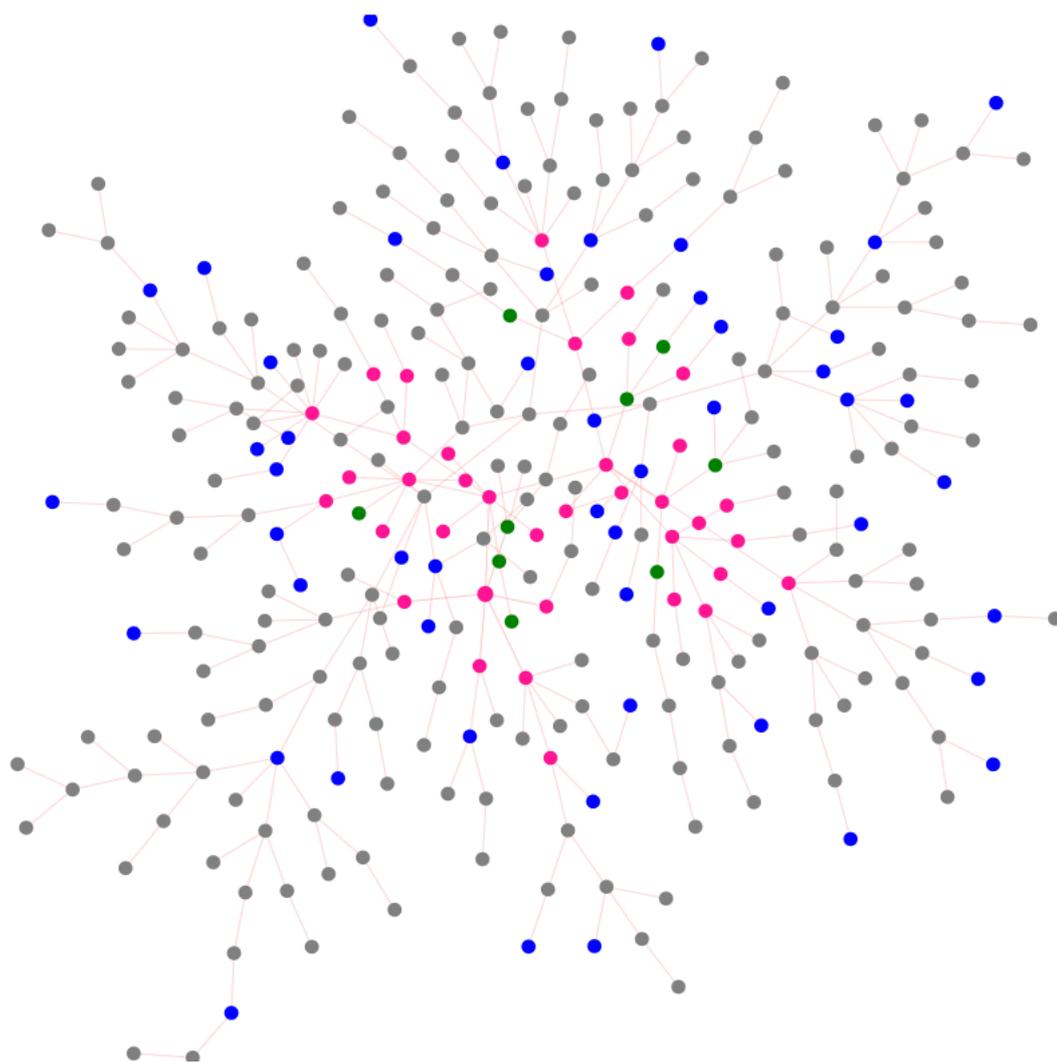


FIGURE 4.10 – Visualisation des nœuds centraux identifiés par les approches heuristique et métaheuristique

La figure ci-dessus illustre les résultats de notre analyse comparative. Les nœuds en rose sont ceux identifiés uniquement par l'approche heuristique, ceux en bleu par la métaheuristique, et ceux en vert par les deux méthodes. Les nœuds en rose, identifiés uniquement par l'heuristique, sont particulièrement visibles au centre du réseau et aux points de jonction clés. Les nœuds en bleu, identifiés par la métaheuristique, sont plus dispersés et ne correspondent pas toujours aux nœuds visuellement centraux. Cette observation souligne la complémentarité des deux approches : l'heuristique est efficace pour identifier les nœuds topologiquement centraux, tandis que la métaheuristique excelle dans la détection de nœuds dont la suppression a un impact significatif sur la connec-

tivité globale du réseau, même s'ils ne semblent pas centraux à première vue.

#### 4.4.1 Comparaison avec des mesures de centralité établies dans la littérature

La comparaison de nos méthodes de détection avec des mesures de centralité bien établies, telles que la centralité de proximité (closeness centrality) et la centralité de degré (degree centrality), offre des perspectives intéressantes sur l'efficacité de nos approches. Pour cette analyse, nous avons utilisé la bibliothèque NetworkX en Python, qui fournit des implémentations robustes et efficaces pour calculer ces mesures de centralité. :

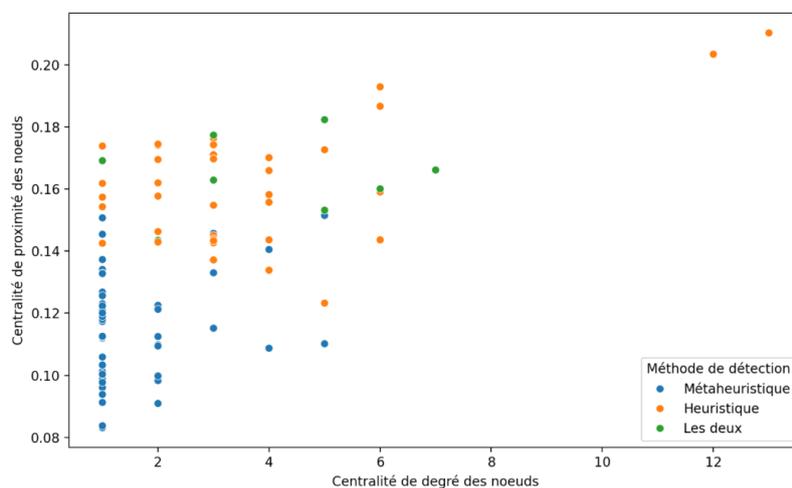


FIGURE 4.11 – Distribution des degrés des noeuds

L'analyse révèle que l'approche heuristique tend à identifier des noeuds avec des valeurs élevées de centralité de proximité, indiquant sa capacité à détecter des noeuds stratégiquement positionnés dans le réseau. En revanche, la méthode métaheuristique semble plus efficace pour détecter des noeuds avec une centralité de degré faible à moyenne, suggérant une sensibilité différente à la structure du réseau.

Cette comparaison met en lumière la complémentarité de nos deux approches par rapport aux mesures classiques de centralité. Alors que l'heuristique semble mieux alignée avec la centralité de proximité, capturant l'importance globale d'un noeud dans le réseau, la métaheuristique offre une perspective différente,

potentiellement en identifiant des nœuds qui, bien que moins centraux selon les mesures classiques, jouent un rôle crucial dans la maintenance de la connectivité du réseau.

Il est important de noter que l'efficacité de l'algorithme de recuit simulé utilisé dans la métaheuristique dépend fortement de sa paramétrisation. Des facteurs tels que la température initiale, le taux de refroidissement et le nombre d'itérations influencent significativement sa capacité à explorer efficacement l'espace des solutions et à identifier les nœuds centraux. Une paramétrisation sous-optimale peut limiter la performance de cette approche, soulignant l'importance d'un ajustement minutieux de ces paramètres pour obtenir des résultats optimaux.

## 4.5 Conclusion

Ce chapitre a présenté une analyse approfondie des résultats de nos approches heuristique et métaheuristique pour l'identification des nœuds centraux dans les réseaux ad hoc. L'approche heuristique s'est montrée efficace pour identifier rapidement les nœuds stratégiques, tandis que la méthode métaheuristique a démontré sa flexibilité en fragmentant significativement le réseau avec peu de suppressions.

La comparaison avec les mesures de centralité classiques a révélé la complémentarité de nos approches. L'heuristique s'aligne bien avec la centralité de proximité, alors que la métaheuristique identifie des nœuds cruciaux pour la connectivité qui ne sont pas nécessairement les plus centraux selon les mesures traditionnelles.

Ces résultats ouvrent des perspectives pour le développement de méthodes hybrides et soulignent l'importance de tester ces approches dans des environnements plus réalistes pour valider leur applicabilité pratique.

# Conclusion générale

---

Ce mémoire a étudié l'identification des nœuds centraux dans les réseaux ad hoc, essentiels pour des domaines variés comme les interventions d'urgence ou les opérations militaires. Deux méthodes ont été comparées : une approche heuristique basée sur la construction d'arborescences et l'analyse de centralité, et une approche métaheuristique utilisant le recuit simulé.

La méthode heuristique s'est révélée rapide et efficace pour les réseaux de taille moyenne, mais limitée face aux structures complexes. L'approche par recuit simulé, plus flexible, a permis d'obtenir des résultats intéressants, notamment en fragmentant significativement le réseau par la suppression d'à peine 20% des nœuds

La comparaison de ces approches a mis en lumière leurs forces et faiblesses respectives. L'heuristique se distingue par sa simplicité et sa rapidité d'exécution, offrant des solutions satisfaisantes dans de nombreux cas. La métaheuristique, quant à elle, brille par sa flexibilité et sa capacité à explorer un plus large éventail de solutions potentielles, bien que cela s'accompagne d'une complexité accrue et d'un temps de calcul plus important.

Les perspectives sont nombreuses. L'exploration de méthodes hybrides, combinant les atouts des deux approches, semble prometteuse. Elle pourrait permettre d'allier la rapidité de l'heuristique à la capacité d'exploration approfondie de la métaheuristique. Par ailleurs, l'apport de l'intelligence artificielle dans l'ajustement dynamique des paramètres pourrait ouvrir de nouvelles voies pour optimiser les performances de ces algorithmes.

Il serait intéressant de tester ces méthodes dans des environnements de simulation plus réalistes, voire sur des réseaux ad hoc réels. Cela permettrait de mieux appréhender leur comportement face aux contraintes du monde réel.

L'étude de la résilience de ces réseaux face aux pannes ou aux attaques constitue également un axe de recherche pertinent.

Il faut garder à l'esprit que cette étude s'est basée sur des simulations. Des tests sur le terrain pourraient révéler des enjeux insoupçonnés. De plus, l'évolution rapide des technologies sans fil pourrait nécessiter une adaptation continue de ces méthodes.

Ce travail a permis d'approfondir notre compréhension de l'optimisation des réseaux ad hoc. Les résultats obtenus ouvrent la voie à de nouvelles recherches et pourraient avoir des applications concrètes dans divers domaines, de la gestion des crises à l'amélioration des réseaux de capteurs, en passant par le renforcement de la sécurité des communications militaires.

# Bibliographie

---

- [1] A.Srinivas, G.Modali, and M.P.Reddy. Node criticality in ad hoc networks. *J Electr Test*, 2011.
- [2] A.S.Tanenbaum and D.J.Wetherall. *Computer Networks (5ème édition)*. Pearson Education, 2011.
- [3] Bernhard Korte and Jens Vygen. *Combinatorial Optimization : Theory and Algorithms*. Springer, 5th edition, 2018.
- [4] Blum Christian and Roli Andrea. *Metaheuristics Network : A New Trends in Optimization*. Springer, 2005.
- [5] Bounceur, Ahcene and Bezoui, Madani and Euler, Reinhardt. *Boundaries and hulls of Euclidean graphs : From theory to practice*. Chapman and Hall/CRC, 2018.
- [6] Chibani Akram and Dolgui Alexandre and Delorme Xavier and Pierreval Henri. Approvisionnement d'une chaîne logistique agile : une approche d'optimisation dynamique. *ARTICLE*, 2014.
- [7] C.K.Toh. *Ad hoc mobile wireless networks : protocols and systems*. Prentice Hall, 2002.
- [8] Clerc Maurice and Siarry Patrick. Une nouvelle métaheuristique pour l'optimisation difficile : la méthode des essaims particulaires. *J3ea*, 3, 2004.
- [9] D.Kempe and F.McSherry. Decentralized approximation of betweenness centrality. *SIAM Journal on Computing*, 39(1) :49–73, 2008.

- [10] El-Ghazali Talbi. *Metaheuristics : From Design to Implementation*. Wiley, 2009.
- [11] E.M.Royer and C.K.Toh. A review of current routing protocols for ad hoc mobile wireless networks. *IEEE Personal Communications*, 1999.
- [12] F.R.K.Chung. *Spectral Graph Theory*. AMS, 1997.
- [13] H.Christian. *Routing in the Internet (2e édition)*. Prentice Hall PTR, 2004.
- [14] Ibrahim H. Osman and James P. Kelly. *Meta-Heuristics : Theory and Applications*. Springer, 1996.
- [15] J.Goshi. Identifying critical nodes in networks. *IEEE Communications Surveys & Tutorials*, 23(2) :1206–1237, 2021.
- [16] J.Hopcroft and R.Tarjan. Algorithm 447 : efficient algorithms for graph manipulation. *Communications of the ACM*, 1973.
- [17] J.Zhu, N.Nakata, and J.JRodrigues. Combined critical node measures for mobile ad hoc networks. *International Journal of Communication Systems*, 2011.
- [18] K.Belkheir and A.Haned. *Réseaux WiFi ad hoc*. PhD thesis, Institut de télécommunication d’Oran, 2008.
- [19] Shafiullah Khan and Jaime Lloret Mauri. *Security for Multihop Wireless Networks*. Taylor Francis Group, 2014.
- [20] K.Y.Lee and M.A.El-Sharkawi. *Modern Heuristic Optimization Techniques : Theory and Applications to Power Systems*, volume 39. John Wiley & Sons, 2008.
- [21] L.Vendramin, A.Campana, and Y.Knyazikhin. Theoretical perspectives on network centralities using a unified approach. *Journal of Complex Networks*, 10(1) :cnab036, 2022.
- [22] M.E.J.Newman. *Networks : An Introduction*. Oxford University Press, 2018.

- [23] M.Hassnaa. *Routage unicast et multicast dans les réseaux mobiles Ad Hoc*. PhD thesis, Ecole Nationale Supérieure des Télécommunications de Paris, 2004.
- [24] Mitchell Melanie. *An Introduction to Genetic Algorithms*. MIT Press, 1998.
- [25] M.Kitsak. Identification of influential spreaders in complex networks. *Nature Physics*, 6(11) :888–893, 2010.
- [26] M.Sedrati, L.Aouragh, L.Guettala, and B.Azeddine. Etude des performances des protocoles de routage dans les réseaux mobiles ad-hoc. 2007.
- [27] N.H.Vaidya. *Mobile Ad Hoc Networks : Routing, MAC and Transport Issues*. Texas AM University, 2000.
- [28] N.Zhang and H.Bao. Wireless network technology and its applications. In *2009 International Conference on Networks Security, Wireless Communications and Trusted Computing*, volume 2, 2009.
- [29] R.Diestel. *Graph Theory*. Springer, 2017.
- [30] Sean Luke. *Essentials of Metaheuristics*. Lulu, 2009.
- [31] S.Ghosh and S.K.Ghosh. Fault-tolerant routing for mobile ad-hoc networks. *International Journal of High Performance Computing and Networking*, pages 235–244, 2008.
- [32] S.Neumayer, G.Zussman, and R.Cohen. Assessing the vulnerability of geographically correlated network node failures. *IEEE MILCOM 2008*, pages 1–6, 2008.
- [33] S.P.Borgatti. Centrality and network flow. *Social Networks*, pages 55–71, 2005.
- [34] Stefano Basagni and Marco Conti and Silvia Giordano and Ivan Stojmenovic. *Mobile ad hoc networking*. The Institute of Electrical and Electronics Engineers, 2004.
- [35] T.N.Dinh, Y.Xuan, M.T.Thai, and E.K.Park. On approximation of new optimization methods for assessing network vulnerability. *IEEE INFOCOM*, 2009.

- [36] T.Opsahl, F.Agneessens, and J.Skvoretz. Node centrality in weighted networks : Generalizing degree and shortest paths. *Social Networks*, pages 245–251, 2010.
- [37] Zbigniew Michalewicz and David B. Fogel. *How to Solve It : Modern Heuristics*. Springer, 2nd edition, 2004.

# Résumé

---

Cette thèse s'intéresse à l'optimisation des réseaux ad hoc, avec un focus particulier sur l'identification des nœuds centraux. Les réseaux ad hoc, de par leur nature décentralisée et auto-organisée, posent des défis uniques en matière de gestion des ressources et d'optimisation des performances.

L'étude se concentre sur la problématique de l'identification des nœuds centraux dans ces réseaux dynamiques, éléments cruciaux pour la connectivité et l'efficacité du routage. La nature changeante de la topologie du réseau rend cette identification particulièrement complexe. La recherche propose deux approches innovantes : une heuristique basée sur la profondeur des arborescences enracinées et une métaheuristique évaluant l'impact de la suppression des nœuds sur la connectivité globale. Une analyse comparative de ces méthodes a été réalisée à l'aide d'un environnement de simulation.

Ce travail contribue à l'avancement des connaissances sur l'optimisation des réseaux ad hoc, ouvrant des perspectives pour améliorer leur performance et leur résilience dans diverses applications.

**Mots clés :** optimisation des réseaux ad hoc, nœuds centraux, heuristique, métaheuristique, connectivité, performance, résilience.

# Abstract

---

This thesis focuses on the optimization of ad hoc networks, with particular emphasis on identifying central nodes. Ad hoc networks, due to their decentralized and self-organizing nature, present unique challenges in resource management and performance optimization.

The study concentrates on the problem of identifying central nodes in these dynamic networks, crucial elements for connectivity and routing efficiency. The changing nature of the network topology makes this identification particularly complex.

The research proposes two innovative approaches : a heuristic based on the depth of rooted trees and a metaheuristic evaluating the impact of node removal on global connectivity. A comparative analysis of these methods was conducted using a simulation environment. This work contributes to advancing knowledge on the optimization of ad hoc networks, opening perspectives to improve their performance and resilience in various applications.

**Keywords :** ad hoc network optimization, central nodes, heuristic, metaheuristic, connectivity, performance, resilience.

