République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة بجاية
Tasdawit n'Bgayet
Université de Béjaïa

Université A. Mira de Béjaïa

Faculté des Sciences Exactes

Département de Recherche Opérationnelle

Mémoire de Master

en

Recherche Opérationnelle

Option : Modélisation Mathématique et Techniques de Décision

Thème

Théorie des Jeux Appliquée à la Sécurité des Réseaux Ad Hoc

présenté par :

BERRI Sara & BOUHADDI Myria

devant le jury composé de :

Président	M^r	**N. KHIMOUM**	**M.A.B**	**U. A/Mira de Béjaia**
Rapporteurs	M^r	**M. S. RADJEF**	**Professeur**	**U. A/Mira de Béjaia**
	M^{me}	**K. ADEL**	**M.A.A**	**U. A/Mira de Béjaia**
Examinateurs	M^{me}	**F. DJOUADI**	**M.A.B**	**U. A/Mira de Béjaia**
	M^r	**F. SEMCHEDINE**	**M.C.B**	**U. A/Mira de Béjaia**

Béjaia 2012.

Remerciements

Louange à Allah, le miséricordieux, sans Lui rien n'aurait pu être possible.

Nous tenons tout d'abord à remercier le P^r M.S. Radjef pour la confiance qu'il nous a témoignée en acceptant de diriger ce mémoire, pour sa patience et aussi pour nous avoir initié au domaine de la recherche.

Un grand merci à M^{me} K. Adel pour ses précieux conseils, sa disponibilité mais aussi pour sa patience, qu'elle trouve ici nos sincères remerciements.

Nous voudrions également remercier le président et les membres de jury d'avoir accepter de juger notre travail et de consacrer leur temps à la lecture et à la correction de ce mémoire.

Notre reconnaissance va à tous ceux qui nous ont apporté leur aide et particulièrement à M^{elle} Belakbir.

Nos remerciements les plus vifs vont tout particulièrement à nos parents et à nos familles.

Dédicaces

Je dédie ce modeste travail :

A *mes très chers parents qui ont toujours répondu présent et qui m'ont soutenu dans chaque travail que j'ai entrepris.*
A *mes grandes sœurs Hafida, Soraya, Sabrina et leur mari Kiki, Sofiane et Smail.*
A *ma petite princesse Dalycia.*
A *mes adorables neveux Ghiles, Toufik, Yacine et Rassim.*
A *ma tante Louisa, son mari Smail et ses deux filles Siham et Lamia.*
A *ma binôme Myria et à toute sa famille.*
A *toute ma famille.*
A *tous mes amis et en particulier Hamid, Dihya, Drifa, Saida, Linda, Selma et Sarah.*

Sara

Je dédie ce modeste travail :

A *mes parents, les deux êtres les plus chers à mon cœur.*
A *Dalina et Lisa, les plus formidables des sœurs.*
A *Fawzi, la plus merveilleuse des personnes.*
A *ma grand mère, la plus adorable des mamies.*
A *Ismahen, Yasmine, Nadia et Sara, les plus exceptionnelles des amies.*

Myria

Table des matières

Liste des tableaux

Table des figures

Introduction générale

Au cours de ces dernières années, le besoin à plus de mobilité et à pouvoir partager ou échanger de l'information à tout moment a fait naître une nouvelle technologie de réseaux mobiles sans fil, pouvant être classés en deux grandes catégories : avec ou sans infrastructure. Dans les réseaux à infrastructure, nous trouvons des stations de base, qui assurent les communications entre les équipements et quant à la seconde catégorie, elle est constituée des réseaux ad hoc. Ces réseaux ne disposent pas infrastructure préexistante et sont formés de nœuds mobiles interconnectés par des liaisons sans fil. Avec leurs caractéristiques particulières, telles que la mobilité et la facilité de mise en place, les réseaux ad hoc sont déployés dans diverses applications comme les opérations de secours ainsi que les applications militaires et tactiques.

Du point de vue de la sécurité, plusieurs facteurs rendent un réseau ad hoc plus vulnérable qu'un réseau filaire. Ces facteurs dépendent essentiellement de la nature mobile de son environnement et de ses topologies dynamiques. En effet, leur sécurité devient une préoccupation importante, car il est fondamental de maîtriser les problématiques de sécurité pour pallier aux diverses vulnérabilités des systèmes et ainsi assurer un bon fonctionnement global du réseau.

Il existe différents travaux pour l'amélioration des méthodes traditionnelles de sécurité des réseaux ad hoc incluant la théorie des jeux, l'une des approches les plus prometteuse pour la modélisation de ce type de réseaux, qui formalise l'interaction stratégique entre des agents rationnels autonomes, qui adopte un nouveau processus qui prend place dans la démarche de modélisation et qui élargit le champ de réflexion. Cette approche s'est imposée donc comme un outil permettant de mieux analyser et étudier le comportement des différentes entités d'un réseau mobile ad hoc.

La théorie des jeux est apparue au début des années 40, c'est une théorie mathématique qui vise à analyser les situations d'interaction entre plusieurs agents rationnels. La rationalité partage la théorie des jeux en deux classes : la théorie des jeux classiques, quand la rationalité résulte de l'intelligence des joueurs et la théorie des jeux évolutionnaires quand la rationalité résulte d'un processus d'apprentissage et d'adaptation.

La théorie des jeux est née "officiellement" en 1944 avec l'ouvrage fondateur "Theory of Games and Economic Behavior" du mathématicien J.Von Neumann et de l'économiste O. Morgenstern. Cette théorie prend comme hypothèse principale la rationalité forte des individus, chaque individu cherche à maximiser ses gains per-

sonnels en prenant en considération le comportement de ses adversaires. La théorie classique cherche à trouver la meilleure solution pour résoudre les conflits. Dans ce cadre, les théoriciens des jeux ont introduit la notion d'équilibre. L'un des équilibres les plus utilisés est l'équilibre de Nash qui peut conduire chaque individu à une situation de non regret, mais elle ne peut pas lui garantir un gain optimal.

La théorie des jeux évolutionnaires s'est développée à la suite des travaux du biologiste John Maynard Smith, dont le but principal consiste à étudier l'évolution dynamique des populations et à analyser le comportement des différents individus. Maynard Smith a également défini les solutions possibles de ce type de jeu en introduisant le concept fondamental de stratégie évolutionnairement stable (Evolutionary Stable Strategy : ESS) et écrit son livre "Evolution and the theory of game" en 1982 [21].

Dans un réseau, un nœud peut contribuer ou pas à sa sécurité, la décision d'un nœud à participer ou pas au mécanisme de sécurité affecte la décision des autres nœuds. Ainsi, l'interaction existant entre les nœuds peut se modéliser sous forme de jeu en prenant en compte la rationalité limitée dont disposent les nœuds. Un jeu évolutionnaire serait plus adéquat afin d'étudier l'évolution du comportement dynamique des nœuds dans un réseau.

L'objectif de ce mémoire est l'étude de la sécurité des réseaux ad hoc notamment dans des situations de conflit entre les nœuds, tout en optimisant leurs ressources. Le premier chapitre sera consacré en une introduction aux réseaux informatiques en général et aux réseaux ad hoc en particulier où nous présenterons les différentes notions qui leurs sont liées et la notion de clusterisation, nous passerons ensuite à la définition de la sécurité informatique et ses objectifs ainsi que les principales attaques et techniques de défense.

Nous enchaînerons avec le second chapitre qui portera sur les principales notions de la théorie des jeux, les concepts de solution et les principaux types de jeux : les jeux Bayésiens, de signalisation et les jeux évolutionnaires avec les notions propres à chacun de ces types.

Quant au troisième chapitre, il sera consacré à une synthèse bibliographique des travaux déjà réalisés ces dernières années sur la modélisation et le traitement de l'interaction entre les attaquants et défenseurs dans les réseaux ad hoc dans l'optique de tirer des modèles de base qui serviront de base de décision.

La contribution essentielle de notre travail sera présentée dans le dernier chapitre. Elle consiste en un modèle permettant de répondre aux besoins des réseaux ad hoc en termes de sécurité. Nous commencerons d'abord par donner une présentation détaillée du modèle où nous appliquerons la théorie des jeux évolutionnaires pour modéliser le comportement stratégique des nœuds, ensuite nous illustrons les résultats de l'évolution de la population et pour finir nous présenterons les résultats de la simulation et nous évaluerons le modèle à base de ces résultats.

1

Réseaux informatiques et concepts de sécurité

1.1 Introduction

Les réseaux informatiques sont nés du besoin de relier des terminaux distants à un site central puis des ordinateurs entre eux et enfin des machines terminales, telles que les stations de travail ou les serveurs. Dans un premier temps, ces communications étaient destinées au transport des données informatiques. Aujourd'hui, l'intégration de la parole téléphonique et de la vidéo est généralisée dans les réseaux informatiques, même si cela ne va pas sans difficulté.

Assurer la sécurité de ces réseaux est de nos jours devenue un problème majeur dans leur gestion, en effet la transmission d'informations sensibles et le désir d'assurer la confidentialité de celles-ci sont un point primordiale dans la mise en place de réseaux informatiques. Dans ce présent chapitre, nous définirons les réseaux informatiques en général et les réseaux ad hoc en particulier, nous enchainerons ensuite par la sécurité informatique, ses objectifs, les terminologies liées à la sécurité ainsi que les différentes attaques et techniques de défense.

1.2 Réseaux informatiques

Un réseau informatique est un ensemble d'éléments matériels reliés entre eux dans le but de permettre aux utilisateurs de partager des ressources et d'échanger des informations sous forme numérique dont l'intérêt est de diminuer les coûts grâce au partage des données et des périphériques.
A l'origine, la connexion entre les différents éléments du réseau se faisait via des câbles et des fils, mais à travers le temps, le besoin à plus de mobilité et à pouvoir

partager ou échanger de l'information à tout moment, en utilisant des dispositifs mobiles, a fait naître une nouvelle technologie des réseaux sans fil.

Aujourd'hui, les réseaux sans fil sont de plus en plus populaires du fait de leur facilité de déploiement. Ces réseaux jouent un rôle crucial au sein des réseaux informatiques. Ils offrent des solutions ouvertes pour fournir la mobilité ainsi que des services essentiels là où l'installation d'infrastructures n'est pas possible. Une grande catégorie de ces réseaux est constituée des réseaux ad hoc que nous présenterons dans la section ci-dessous.

1.2.1 Réseaux ad hoc

Un réseau mobile ad hoc, appelé aussi MANET (Mobile Ad hoc NETwork), est un ensemble autonome et coopératif de nœuds mobiles qui se déplacent et communiquent de manière autonome par une transmission sans fil appelée ondes radio qui ne suppose pas d'infrastructure préexistante.

Un nœud peut à la fois communiquer directement avec d'autres nœuds ou servir de relais. Un relais permet à des nœuds se trouvant hors de leur rayon de transmission [1] les uns des autres de communiquer. Ces réseaux sont dits ad hoc dans la mesure où ils ne nécessitent pas d'infrastructure fixe. Ils peuvent exister temporairement pour répondre à un besoin ponctuel de communication.

1.2.1.1 Modèle mathématique pour les réseaux 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 (les unités ou les hôtes [2] mobiles) du réseau et E_t modélise l'ensemble des connexions qui existent entre ces nœuds à l'instant t. Si $e = (u, v)$ appartient à E_t, cela signifie que les nœuds u et v sont en mesure de communiquer directement à l'instant t.

1. rayon de transmission ou bien portée radio : est une zone bien spécifique à chacun des nœuds du réseau dans laquelle une transmission directe est possible

2. Hôte : est une machine client connectée à un réseau.

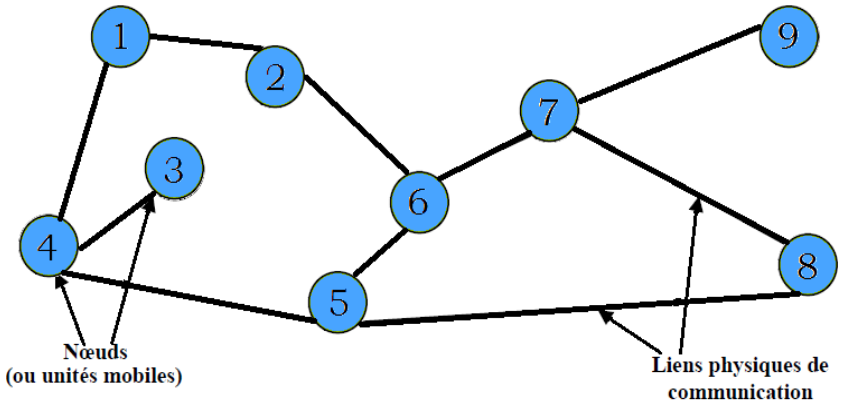


FIGURE 1.1 – La modélisation d'un réseau ad hoc

1.2.1.2 Propriétés et spécificités des réseaux ad hoc

Les spécificités des réseaux sans fil ad hoc sont multiples. Nous pouvons les répartir en six grands thèmes traitant des caractéristiques des nœuds, de la gestion de l'énergie, des caractéristiques du réseau, de la mobilité et de la configuration de la sécurité.

- **Energie limitée des nœuds :** les nœuds mobiles disposent d'une énergie limitée par la capacité de leur batterie qui est difficilement rechargeable en cours de déploiement. Pour prolonger la durée de vie du réseau il est alors nécessaire de chercher à réduire la consommation d'énergie.

- **Topologie dynamique et mobilité :** une caractéristique particulière aux réseaux ad hoc est la variation de leur topologie. Les unités mobiles du réseau, se déplacent d'une façon libre et arbitraire. Par conséquent la topologie du réseau peut changer, à des instants imprévisibles, d'une manière rapide et aléatoire. Les liens de la topologie peuvent être unidirectionnels ou bidirectionnels.

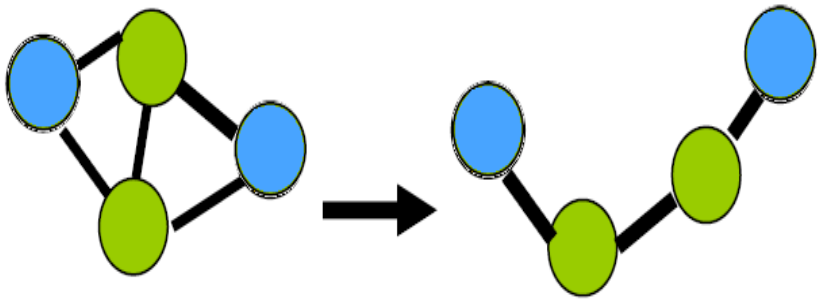


FIGURE 1.2 – Le changement de la topologie des réseaux ad hoc

- **Absence d'infrastructure :** les réseaux ad hoc se distinguent des autres réseaux mobiles par la propriété d'absence d'infrastructures préexistantes et de

tout genre d'administration centralisée. Les hôtes mobiles sont responsables d'établir et de maintenir la connectivité du réseau d'une manière continue.

- **Routage multi-sauts :** cela signifie que des communications entre deux nœuds doivent pouvoir s'effectuer même si ceux-ci sont hors de portée de communication directe. La connaissance réciproque de leur existence et les échanges d'information doivent être possibles en traversant d'autres nœuds du réseau.
- **Chaque nœud du réseau est à la fois hôte et routeur :** chaque nœud du réseau contribue au bon acheminement des informations dans le réseau. Pour ce faire, chaque nœud essaie de posséder une connaissance partielle du réseau la plus étendue possible pour jouer le rôle de routeur pour lui-même et pour tout autre nœud qui le lui demandera. Cette connaissance partielle se manifeste par la connaissance des voisins immédiats et leurs positions en fonction du temps.
- **Rapidité de déploiement :** les réseaux ad hoc peuvent être facilement installés dans les endroits difficiles à câbler, ce qui élimine une bonne part du travail et du coût généralement liés à l'installation et réduit d'autant le temps nécessaire à la mise en route.
- **Sécurité physique limitée :** les réseaux ad hoc sont plus touchés par le paramètre de sécurité que les réseaux filaires classiques. Cela se justifie entre autres par les vulnérabilités des liens radio aux attaques, ainsi que les contraintes et limitations physiques qui font que le contrôle des données transférées doit être minimisé.

1.2.1.3 Applications des réseaux ad hoc

La particularité d'un réseau ad hoc est qu'il n'a besoin d'aucune installation fixe, ce qui lui permet d'être rapide et facile à déployer. Les opérations tactiques comme les opérations de secours, militaires ou d'explorations trouvent en ad hoc, le réseau idéal. La technologie ad hoc intéresse également la recherche des applications civiles. Nous distinguons entre autres :

- Les services d'urgence : opération de recherche et de secours des personnes, tremblement de terre, feux, inondation, dans le but de remplacer l'infrastructure filaire ;
- Le travail collaboratif et les communications dans des entreprises ou bâtiments : dans le cadre d'une réunion ou d'une conférence par exemple ;
- Applications commerciales : pour un paiement électronique distant (taxi) ou pour l'accès mobile à l'internet, ou servir de guide en fonction de la position de l'utilisateur ;
- Réseaux de capteurs : pour des applications environnementales (climat, activité de la terre, suivi des mouvements des animaux,...) ou domestiques (contrôle des équipements à distance). Ces réseaux sont composés d'un grand nombre de petits équipements autonomes dans lesquels, chaque capteur peut sentir, calculer, envoyer/recevoir des données pour rassembler des informations dans le but de satisfaire une tâche concrète.

Les applications potentielles des réseaux ad hoc sont nombreuses. Par exemple, nous pouvons penser qu'un groupe de personnes avec des ordinateurs portables lors d'une conférence qui souhaite échanger des fichiers peut rapidement mettre en place un réseau ad hoc sans avoir recours à une infrastructure supplémentaire. Les réseaux ad hoc sont idéals dans des zones où un tremblement de terre ou d'autres catastrophes naturelles ont détruit les infrastructures de communication [15].

D'une façon générale, les réseaux ad hoc sont utilisés dans toute application où le déploiement d'une infrastructure réseau filaire est trop contraignant, soit parce qu'il est difficile à mettre en place, soit parce que la durée d'installation du réseau ne justifie pas de câblage à demeure.

1.2.1.4 Concept de Clusterisation

Un réseau ad hoc, étant classiquement considéré comme un réseau non structuré c'est à dire tous les nœuds du réseau ont des rôles égalitaires, doit être organisé pour faciliter son utilisation. Ainsi, la structuration est une approche importante pour simplifier le fonctionnement d'un réseau ad hoc. Pour cela plusieurs approches ont été proposées dans la littérature qui se basent essentiellement sur la technique de clusterisation que nous définirons ci-dessous.

Définition 1.2.1 *La Clusterisation consiste à partitionner le réseau en groupes d'entités, appelés* ***clusters****, plus homogènes selon une métrique spécifique ou une combinaison de métriques. Les clusters ne sont pas nécessairement disjoints, chaque cluster est identifié par un nœud particulier appelé* ***cluster head****, sorte de centre du cluster, qui agit comme un coordinateur local dans son cluster. Le choix du cluster head se fait sur la base d'une métrique bien définie.*

Avanatages de la clusterisation

Comme nous l'avons vu, le principe de la clusterisation consiste à organiser le réseau en une structure hiérarchique. Cette technique de structuration du réseau possède des avantages, nous en citons alors :

- Les nœuds de chaque cluster sont supervisés par leur cluster head qui peut coordonner l'accès au canal, épargnant ainsi les ressources gaspillées dans la retransmission due aux collisions ;
- Permet de ne stocker que des informations partielles du réseau ;
- Optimiser les dépenses en énergie.

1.2.1.5 Algorithmes de Clusterisation

Plusieurs mécanismes de clusterisation ont été proposés dans la littérature. Les clusters sont identifiés par leur cluster head. Les différents algorithmes se distinguent sur le critère de sélection des cluster heads, c'est à dire la métrique. Dans cet axe, certains mécanismes de clusterisation ont choisi des critères simples, comme l'identifiant (ID), le degré. Alors que d'autres approches ont adopté des sélections plus élaborées

en s'appuyant sur une combinaison de critères afin de sélectionner les nœuds les plus appropriés pour assurer les fonctionnalités du chef de groupe.

Algorithme basé sur le plus petit identifiant (ID)

Ephremides, Weiselthier et Baker ont proposé dans [3] l'un des premiers algorithmes de clusterisation pour les réseaux ad hoc. Il s'agit de l'algorithme de plus petit ID appelé aussi Linked Cluster Architecture (LCA). Chaque nœud se déclare cluster head ou non en se basant sur son identifiant et ceux de ses voisins. Un nœud peut être dans l'un de ces quatre statuts : ordinaire, cluster head, membre ou passerelle. Initialement, tous les nœuds ont un statut de nœud ordinaire. La formation des clusters suit les règles suivantes :

Algorithm 1 Algorithme du plus petit ID

1. Si un nœud u possède le plus petit identifiant dans son voisinage à un saut, il se déclarera comme cluster head et ses voisins à un saut dont les identifiants sont supérieurs au sien le joignent et deviennent des nœuds membres ;
2. Si non, il attendra que tous ses voisins à un saut déclareront leurs statuts. Ainsi si un parmi eux se déclare cluster head alors le nœud u déclare à son voisinage à un saut son statut de nœud membre ;
3. Une fois que tous les nœuds ont soit le statut de membre ou de cluster head, alors si un nœud a parmi ses voisins à un saut plus qu'un cluster head, il se déclarera nœud passerelle.

L'algorithme LCA construit à la fois des clusters recouvrants [3] et non-recouvrants.

Algorithme basé sur le degré le plus élevé

Gerla et Tsai ont proposé dans [9] un algorithme de clusterisation appelé High-Connectivity Clustering (HCC). Cet algorithme se base sur l'élection du nœud dont le degré est le plus élevé. Le degré d'un nœud se calcule en fonction de sa distance par rapport aux autres. Les différentes phases de cet algorithme sont les suivantes :

Algorithm 2 Algorithme HCC

1. Chaque nœud diffuse son ID aux nœuds qui se trouvent à sa portée de transmission (ses voisins) et le nœud avec un nombre maximum de voisins, c'est à dire avec un degré maximal, est choisi comme cluster head et ses voisins deviennent membres de ce cluster et ne peuvent plus participer au processus électoral ;
2. Le processus continue jusqu'à ce qu'il n'y ait plus de nœuds à affilier aux clusters.

Nous aurons à la fin de ce processus, deux nœuds d'un cluster sont à deux sauts et le chef du cluster est directement lié à chacun de ses membres [1].

3. C'est à dire qu'il existe des nœuds du réseau qui appartiennent à 2 clusters simultanément

L'avantage de cet algorithme, est qu'il génère un nombre réduit de clusters puisqu'il favorise les nœuds ayant le plus fort degré pour être cluster heads i.e. les nœuds qui couvrent plus de nœuds voisins. Dans un environnement mobile, cet algorithme produit des cluster heads qui ne sont pas susceptibles de jouer leur rôle de chef pour très longtemps puisque leurs degrés changent très fréquemment, contrairement à l'algorithme du plus petit ID où les nœuds de faible identifiant ont tendance à garder le statut de cluster head plus longtemps.

Algorithme CONID

Dans le but de tirer profit des avantages de ces deux algorithmes "plus petit ID" et "plus grand Degré", Chen et Stojmenović [6] ont proposé de combiner les deux algorithmes en un seul algorithme appelé "CONID". Cet algorithme considère le degré (connectivité) des nœuds comme clé primaire et l'identifiant ID des nœuds comme clé secondaire pour choisir les cluster heads. Dans "CONID", à chaque nœud u du réseau est associée une paire :

$$CONID_u = (CON_u, ID_u).$$

Cette paire indique la connectivité CON_u et l'identifiant ID_u d'un nœud u. Si deux nœuds u et v ont respectivement les paires :

$$CONID_u = (CON_u, ID_u) \text{ et } CONID_v = (CON_v, ID_v),$$

alors le choix du cluster head entre ces deux nœuds est illustré par le schéma algorithmique suivant :

Algorithm 3 Algorithme CONID

Si $(CON_u > CON_v) \vee (CON_u{=}CON_v \wedge ID_u < ID_v)$
alors "u est cluster head"
Sinon "v est cluster head"

Cet algorithme génère aussi des clusters recouvrants et les nœuds qui appartiennent à plus d'un cluster sont considérés comme des nœuds passerelles.

Algorithme basé sur le poids de nœuds

Les approches que nous avons présentées jusque là se basent sur le choix d'une seule métrique pour l'élection du cluster head, ce choix n'étant pas très judicieux pour la stabilité des clusters formés, nous présenterons une technique qui repose sur le poids du nœud qui indique sa compétence au rôle de cluster head. En générale, le poids du nœud est représenté par une somme pondérée de quelques métriques, jugées nécessaire pour la stabilité de la topologie du réseau [8]. Comme l'indique la formule suivante :

$$Weight(u) = \sum_{i=1}^{k} \alpha_i P_i \quad avec \quad \sum_{i=1}^{k} \alpha_i = 1.$$

où k est le nombre de métriques, α_i est le coéficient de la métrique et P_i la valeur de la métrique.

L'algorithme Weighted Clustering Algorithm (WCA) [5] implique quatre métriques dans le calcul du poids d'un nœud : la différence de degré D_u, la somme des distances avec ses voisins P_u, la mobilité relative moyenne M_u et le temps de service en tant que cluster head T_u tel que :

$$Weight(u) = \alpha D_u + \beta P_u + \gamma M_u + \delta T_u,$$

$$avec \quad \alpha + \beta + \gamma + \delta = 1.$$

- D_u, la différence de degré, est la différence entre le degré du nœud u et une valeur δ qui représente le nombre de nœuds qu'un cluster head peut servir. Pour calculer la valeur de δ, les auteurs dans [5] supposent que le nombre de nœuds qu'un cluster head doit servir est déterminé à l'avance, sans toutefois donner des détails sur la façon dont cette valeur est calculée ;
- P_u est un paramètre qui est défini comme la somme des distances entre un nœud et ses voisins. Ce facteur est lié à la consommation d'énergie car plus la communication est à grande distance plus la puissance nécessaire est importante, de ce fait l'énergie consommée est grande.
- M_u, La mobilité relative, est calculée à partir de la vitesse moyenne de chaque nœuds pendant un délai déterminé T ;
- T_u est un paramètre qui représente le temps cumulé d'un nœud étant cluster head. Il mesure la quantité d'énergie qui a été consommée de la batterie. Un cluster head consomme plus d'énergie qu'un nœuds ordinaire car il a des responsabilités supplémentaires.

Le processus d'élection de cluster head est le suivant :

Algorithm 4 Algorithme basé sur le poids de nœuds

1. Le nœud ayant le plus petit poids dans son voisinage devient cluster head et ses voisins le joignent et forment un cluster.
2. L'algorithme se termine une fois que tous les nœuds du réseau se soient répartis en cluster, c'est à dire que tous les nœuds sont soient cluster heads ou bien membres.

La distance entre les membres d'un cluster doit être inférieur ou égal à l'intervalle de transmission entre eux et deux cluster heads ne peuvent être voisins directs [1].

Nous allons maintenant aborder la sécurité informatique, les attaques et vulnérabilités que nous pouvons retrouver dans les réseaux.

1.3 Sécurité informatique

La sécurité informatique est l'ensemble de méthodes, techniques et outils mis en œuvre pour protéger les systèmes et réduire leur vulnérabilité contre les menaces

accidentelles et les actions malveillantes [10].

1.3.1 Objectifs et principaux services de la sécurité informatique

La sécurité informatique est l'une des questions les plus importantes liées à l'internet. Elle empêche la divulgation et la modification non-autorisée de données, ainsi que l'utilisation non-autorisée de ressources réseau ou informatiques de façon générale.

Pour remédier aux failles et pour contrer les attaques, la sécurité informatique se base sur un certain nombre de services qui permettent de mettre en place une réponse appropriée à chaque menace. Décrivons les cinq principaux services de sécurité :

1. **Intégrité :** est de garantir que les données sont bien celles que l'on croit être. Vérifier l'intégrité des données consiste à déterminer si les données n'ont pas été altérées durant la communication. Autrement dit, c'est de protéger ces données contre les menaces qui peuvent causer leur modification non autorisée.
2. **Confidentialité :** consiste à assurer que seules les personnes autorisées aient accès aux ressources échangées ; c'est-à-dire rendre l'information intelligible qu'aux deux entités de la transaction et la protéger contre les menaces pouvant causer sa divulgation non autorisée.
3. **Disponibilité :** la disponibilité vise à assurer que le système soit bien prêt à l'emploi. Elle le protège également contre les menaces qui peuvent causer sa perturbation, y compris la non disponibilité des services, le vol des données et la destruction du matériel. Elle veille aussi à ce que les services soient accessibles et que l'accès au système par des sujets non autorisés soit prohibé.
4. **Non répudiation :** permet de garantir qu'aucun des correspondants ne pourra nier la transaction. C'est à dire indiquer que des actions ou bien des évènements ont bien eu lieu.
5. **Authentification :** consiste à assurer l'identité d'un utilisateur, c'est-à-dire de garantir à chacun des correspondants que son partenaire est bien celui qu'il croit être. Un contrôle d'accès peut permettre, par exemple en utilisant un mot de passe, l'accès à des ressources uniquement aux personnes autorisées.

1.3.2 Terminologie de la sécurité informatique

La sécurité informatique utilise des termes propres à elle. De manière à bien comprendre ce rapport, il est nécessaire d'en définir certains :

- **Vulnérabilité :** dans le domaine de la sécurité informatique, une vulnérabilité ou faille est une faiblesse dans un système informatique, permettant à un attaquant de porter atteinte à l'intégrité de ce système, c'est-à-dire à son fonctionnement normal, à la confidentialité et l'intégrité des données qu'il contient.

Ces faiblesses conduisent à l'exploitation des ressources informatiques par des menaces dans le but de les compromettre et cela peut causer des pertes importantes.
Ces vulnérabilités sont la conséquence des imperfections dans la conception, la mise en œuvre ou l'utilisation d'un composant matériel ou logiciel du système.

- **Les attaques :** une attaque est l'exploitation d'une vulnérabilité d'un système informatique susceptible de lui causer des dommages. Il peut y avoir plusieurs attaques pour une même vulnérabilité mais toutes les vulnérabilités ne sont pas exploitables.
  Les attaques peuvent à première vue être classées en 2 grandes catégories :
  - **Attaques passives :** consistent à écouter sans modifier les données ou le fonctionnement du réseau. Elles sont généralement indétectables mais une prévention est possible.
  - **Attaques actives :** consistent à modifier des données ou des messages, à s'introduire dans des équipements réseau ou à perturber le bon fonctionnement de ce réseau. Noter qu'une attaque active peut être exécutée sans la capacité d'écoute. De plus, il n'y a généralement pas de prévention possible pour ces attaques, bien qu'elles soient détectables.
- **Contre-mesures :** ce sont les procédures ou techniques permettant de résoudre une vulnérabilité ou de contrer une attaque spécifique.
- **Menaces :** ce sont des adversaires déterminés et capables de monter une attaque en exploitant une vulnérabilité.
- **Intrusions :** on appelle intrusion l'ensemble des actions non autorisées ou l'abus d'utilisation d'un système informatique qui ont pour but de compromettre l'intégrité, la confidentialité ou la disponibilité d'une ressource. Ces actions de franchissement d'un accès non-autorisé ou de manipulation d'une ressource, peuvent être menées par un individu externe n'ayant aucun privilège sur les ressources d'un système, ou par un individu interne qui outrepasse ses privilèges.

1.4 Principales attaques

Un système informatique peut être la cible de plusieurs attaques, elles sont si nombreuses qu'il serait illusoire de prétendre les décrire toutes. Il est cependant possible, afin de mieux appréhender ces attaques, de connaître les principales catégories d'attaques, qui ont pour point commun d'exploiter des faiblesses de sécurité. Cette section sera réservée à la présentation des différentes catégories d'attaques.

1.4.1 Attaques d'accès

Une attaque d'accès est une tentative d'accès à l'information par une personne non autorisée. Ce type d'attaque concerne la confidentialité de l'information. Nous allons alors définir quelques une de ces attaques :

- **Sniffing :** cette attaque est utilisée par les pirates informatiques pour obtenir des mots de passe. Grâce à un logiciel appelé *renifleur de paquets (sniffer)*, nous pouvons intercepter tous les paquets qui circulent sur un réseau même ceux qui ne nous sont pas destinés.
- **Cheval de Troie :** ce programme malveillant est un programme informatique caché dans un autre, après avoir accédé au système, ce programme installe un logiciel qui va, lui transmettre par internet les informations des disques durs. Un tel logiciel, aussi appelé *troyen* ou *trojan*, peut aussi être utilisé pour générer de nouvelles attaques sur d'autres serveurs en passant par la machine déjà infectée.
- **Porte dérobée :** lorsqu'un pirate informatique arrive à accéder à un serveur à l'aide d'une des techniques présentées précédemment, il souhaiterait y retourner sans avoir à tout recommencer. Pour cela, il laisse donc des portes dérobées (backdoor) qui lui permettrons de reprendre facilement le contrôle du système informatique, à titre d'exemple, la création d'un nouveau compte administrateur avec un mot de passe choisi par le pirate.
  Dans tous les cas, l'administrateur perd le contrôle total du système informatique. Le pirate peut alors récupérer les données qu'il souhaite, voler des mots de passe ou même détruire des données.

1.4.2 Attaques de modification

Une attaque de type modification consiste, pour un attaquant à tenter de modifier des informations. Ce type d'attaque est dirigé contre l'intégrité de l'information et les plus connues sont :

- **Virus :** les virus sont des programmes qui s'exécutent et se répliquent automatiquement. Ils modifient le fonctionnement d'un ordinateur sans que l'utilisateur ne s'en aperçoive ni ne l'autorise. Lorsqu'ils sont actifs, les virus peuvent endommager des fichiers, engendrer un comportement imprévisible du système ou afficher des messages inopportuns. Ceux-ci se distinguent des chevaux de troie, des vers et avec autres programmes de contamination par leur capacité à se répliquer de façon autonome.
- **Vers (Worm) :** les vers sont des petits programmes informatiques parasites, autonomes et capables de se propager à travers le réseau et à travers la mémoire des ordinateurs infectés, leur but est de grignoter des ressources système : CPU, mémoire, espace disque, bande passante.... Ils sont capables de se propager à l'intérieur de la mémoire d'un ordinateur passant d'un système à l'autre grâce au réseau informatique, flash disc ou autre. Après avoir pénétré au sein du système de leur victime, ces vers se dupliquent tout en se camouflant, saturant ainsi la mémoire disponible, ce qui implique le ralentissement du système ou son blocage et nous pouvons avoir même le crash du système victime et aussi les pertes de données.
  Les vers sont totalement différents des virus, car ils ne cherchent pas à altérer le contenu de l'ordinateur. Leur fonctionnement peut s'assimiler à celui d'un

parasite vivant, ils cherchent simplement à survivre, pour cela ils doivent se propager sur les ordinateurs accessibles.

1.4.3 Attaques par saturation (déni de service DoS)

Les attaques par saturation sont des attaques informatiques qui consistent à envoyer des milliers de messages depuis des dizaines d'ordinateurs, dans le but de submerger les serveurs d'une société, de paralyser pendant plusieurs heures son site web et d'en bloquer ainsi l'accès aux internautes. Cette technique de piratage assez simple à réaliser est jugée comme de la pure malveillance. Elle ne fait que bloquer l'accès aux sites, sans en altérer le contenu. Parmi les différents types d'attaques par DoS, il existe en particulier :

- **Flooding :** cette attaque consiste à envoyer à une machine de nombreux paquets IP de grosse taille, à condition d'avoir un ping (le temps que met une information pour faire un aller retour entre deux machines) très court. La machine cible ne pourra donc pas traiter tous les paquets et finira par se déconnecter du réseau. Pour l'éviter, une solution consiste à ne pas divulguer son adresse IP.
- **Smurf :** le smurf est une attaque qui s'appuie sur le ping et les serveurs de broadcast(diffusion). Cette attaque consiste à se faire passer pour la machine cible en falsifiant son adresse IP.
- **Débordement de tampon :** cette attaque se base sur une faille du protocole IP. Nous envoyons à la machine cible des données d'une taille supérieure à la capacité d'un paquet. Celui-ci sera alors fractionné pour l'envoi et rassemblé par la machine cible. A ce moment, il y aura débordement des variables internes. Suite à ce débordement, plusieurs cas se présentent : la machine se bloque ou se redémarre.

1.4.4 Attaques de répudiation

La répudiation est une attaque contre la responsabilité. Autrement dit, la répudiation consiste à tenter de donner de fausses informations ou de nier qu'un événement ou une transaction se soit réellement passé. Il existe en particulier :

- **Spoofing IP :** le spoofing IP consiste en une usurpation, par un utilisateur du réseau, d'une adresse IP afin de se faire passer pour une autre machine en falsifiant son adresse IP. Cette technique repose sur les liens d'authentification et d'approbation qui existent au sein d'un réseau.

1.5 Techniques de défense et protocoles de sécurité

Comme nous l'avons déjà présenté précédemment, les réseaux informatiques sont potentiellement susceptibles d'être attaqués à tout moment. Et afin de se prémunir

contre une utilisation des réseaux qui viserait à s'approprier indûment des informations, il est nécessaire de connaître certaines techniques de sécurité informatique. Dans cette partie du rapport, nous donnerons les plus utilisés.

1.5.1 Techniques de défense

Les techniques de défense désignent l'ensemble des lois et des consignes afin de protéger les ressources et les informations contre tout préjudice à leur confidentialité, leur intégrité et leur disponibilité, lequel serait dû à un usage inapproprié.

1.6.1.1 Politique de sécurité

Une politique de sécurité est un ensemble de règles et pratiques qui régissent la façon dont l'information sensible et les autres ressources sont gérées, protégées et distribuées à l'intérieur d'un système spécifique. Sa mise en œuvre se fait selon les quatre étapes suivantes :

- Identifier les besoins en terme de sécurité, les risques informatiques pesant sur l'entreprise et leurs éventuelles conséquences ;
- Elaborer des règles et des procédures à mettre en œuvre dans les différents services de l'organisation pour les risques identifiés ;
- Surveiller et détecter les vulnérabilités du système d'information et se tenir informé des failles sur les applications et matériels utilisés ;
- Définir les actions à entreprendre et les personnes à contacter en cas de détection d'une menace.

1.6.1.2 Cryptographie

La cryptographie est une des disciplines de la cryptologie utilisée dans la sécurité informatique, s'attachant à protéger des messages assurant ainsi la confidentialité, l'authenticité et l'intégrité en s'aidant souvent de secrets ou clès. Elle est basée sur deux principes dont le premier est le **chiffrement** qui a comme but de transformer un message clair en un message incompréhensible appelé *cryptogramme*. Le second est le **déchiffrement** qui consiste en l'opération inverse du chiffrement.
Il existe deux manières différentes de cryptage (chiffrement), le cryptage symétrique, et le cryptage asymétrique.

- **Cryptage symétrique :** est basé sur l'utilisation d'une clé partagée entre deux parties communicantes, cette clé sert à crypter et décrypter les messages ;
- **Cryptage asymétrique :** chaque utilisateur dispose d'un jeu unique de clés, dont l'une est privée (secrète) et l'autre publique. Pour recevoir des documents protégés, le détenteur d'un jeu de clés envoie sa clé publique à ses interlocuteurs, qui l'utilisent pour chiffrer les données avant de les lui envoyer. Seul le destinataire et détenteur des clés peut lire les informations en associant sa clé privée à sa clé publique.

1.6.1.3 Proxy

Un serveur proxy (mandataire en français) est un serveur informatique qui a pour fonction de relayer des requêtes entre un poste client et un serveur. Les serveurs proxys sont notamment utilisés pour assurer les fonctions suivantes :

- La journalisation des requêtes ;
- La sécurité du réseau local ;
- Le filtrage et l'anonymat.

1.6.1.4 Antivirus

Un antivirus est un logiciel conçu pour repérer les traces d'activité des virus, les bloquer et isoler ou supprimer les fichiers qui en sont responsables. Leur mode de fonctionnement est basé sur une veille permanente, à deux niveaux :

- Installation et activation d'un programme antivirus sur chaque ordinateur ;
- Mise à jour de l'antivirus : la surveillance par l'antivirus se réfère à une base de données contenant les signes d'activité de tous les virus connus. Chaque jour, de nouveaux virus apparaissent, inventés par des experts en programmation désireux d'éprouver leurs compétences ; en permanence, d'autres experts surveillent l'apparition de ces nouveaux programmes et conçoivent des antidotes. Nous comprenons qu'un antivirus ne sera efficace que s'il est régulièrement actualisé, pour détecter les manifestations de tous les nouveaux virus.

1.6.1.5 Firewall (Pare-feu)

C'est un système ou un groupe de systèmes qui gère les contrôles d'accès entre deux réseaux et permet de réduire les possibilités d'attaque à distance. Deux mécanismes sont utilisés : le premier consiste à interdire le trafic, et le deuxième à l'autoriser. Un système pare-feu contient un ensemble de règles prédéfinies permettant :

- D'autoriser la connexion (allow) ;
- De bloquer la connexion (deny) ;
- De rejeter la demande de connexion sans avertir l'émetteur (drop).

1.6.1.6 VPN (Virtual Private Network)

Un réseau virtuel privé (Virtual Private Network, VPN) est une technique permettant à un ou plusieurs postes distants de communiquer de manière sûre, tout en empruntant les infrastructures publiques. Ce réseau consiste en la fabrication d'un tunnel logique qui sera contracté par les communications de l'entreprise, lesquelles seront véhiculées dans cette tranchée numérique construite sur un réseau fréquenté par d'autres usagers. Un VPN est donc une communication sécurisée entre deux points d'un réseau public, d'où l'expression de tunnel.

1.6.1.7 IDS (Intrusion Detection System)

Un système de détection d'intrusions est un processus de surveillance des événements se trouvant dans un système d'ordinateurs ou du réseau, il permet de détecter en temps réel et de façon continue des tentatives d'intrusion.
Une fois que l'IDS est installé et configuré, il interagit avec l'environnement dans lequel il est implanté de façon à pouvoir le protéger d'éventuelles intrusions. Durant la phase de détection, il capte l'information via une source de donnée, il l'analyse et il transmet les résultats de l'analyse à l'opérateur qui agit en conséquence, selon qu'il y ait ou pas d'attaques.

Il existe deux types de systèmes de détection d'intrusion :

- **Systèmes de détection des intrusions de type hôte (HIDS) :** ces systèmes sont en fait les premiers systèmes mis en œuvre pour la détection d'intrusion. Ils sont installés sur une machine hôte pour la protéger.
- **Systèmes de détection des intrusions réseaux (NIDS) :** ces systèmes sont mis en œuvre pour la protection des réseaux. Ils se basent sur le principe de l'analyse des paquets transitant dans le réseau pour déterminer si une attaque a lieu.

1.5.2 Protocoles de sécurité

Un certain nombre de protocoles sont utiles pour la sécurité d'un réseau informatique, et dans cette partie, nous en définirons les plus importants.

1.6.2.1 S-HTTP (Secure HTTP)

S-HTTP est un procédé (protocole) de sécurisation des transactions HTTP reposant sur une amélioration du protocole HTTP mis au point en 1994 par l'EIT (Entreprise Intégration Technologies). Il permet au visiteur de vérifier l'identité du site auquel il accède grâce à un certificat d'authentification émis par une autorité tierce réputée fiable. Il garantit théoriquement la confidentialité et l'intégrité des données envoyées par l'utilisateur et reçues du serveur.
S-HTTP est généralement utilisé pour les transactions financières en ligne : commerce électronique, banque en ligne, courtage en ligne... Il est aussi utilisé pour la consultation de données privées, comme les courriers électroniques par exemple.

1.6.2.2 SSL (Secure Sochet Layer)

Le SSL (Secure Socket Layer) est un protocole développé par *Netscape* en 1996 qui est rapidement devenu la méthode par excellence pour sécuriser les transmissions de données via internet. Le protocole SSL peut être divisé en deux sous protocoles : l'encodage (record), définit tous ce qui est lié à l'envoi des données. Et la négociation (handshake) est utilisée pendant la phase de négociation entre le client et le serveur jusqu'à ce que tous les paramètres soient validés par l'un et l'autre.

1.6.2.3 IPsec (IP secure)

IPsec (Internet Protocol Security) est un ensemble de protocoles utilisant des algorithmes permettant le transport de données sécurisées sur un réseau IP. Son objectif est d'authentifier et de chiffrer les données : le flux ne pourra être compréhensible que par le destinataire final (chiffrement) et la modification des données par des intermédiaires ne pourra être possible (intégrité). IPsec est souvent un composant de VPN, il est à l'origine de son aspect sécurité.

Les réseaux ad hoc étant par définition sans infrastructure, ils ne peuvent pas bénéficier de tous les services de sécurité présentés.

1.6 Conclusion

Ce chapitre s'est consacré en premier lieu à la définition des réseaux informatiques en général et les réseaux sans fil ad hoc en particulier, où nous avons introduit leurs caractéristiques et les notions qui leurs sont propres. Nous avons par la suite présenté quelques concepts de sécurité informatique et mis en avant certains risques auxquels elle est exposée, ainsi qu'une multitude de techniques de défense. Pour les réseaux ad hoc, l'application de certains mécanismes de sécurité développés pour les réseaux filaires est délicate, voir impossible, en raison de leur caractéristiques particulières qui font d'eux des réseaux à part.

Toutes ces contraintes concourent à rendre la sécurité des réseaux ad hoc difficile et complexe à appréhender. Afin de remédier à cette difficulté, des travaux introduisent la théorie des jeux qui apparaît comme une solution plausible et efficace.

Dans le prochain chapitre, nous passerons en relief les principales notions de la théorie des jeux, l'une des modélisations les plus prometteuse pour la sécurité des réseaux ad hoc.

2
Sur la théorie des jeux

2.1 Introduction

La théorie des jeux consiste à étudier les situations de conflits qui peuvent exister entre des agents en interaction. Elle a connu une véritable explosion au cours de ces dernières années aussi bien sur le plan théorique qu'au niveau des applications. Elle est devenue un outil central dans plusieurs disciplines comme la biologie, le transport routier et les réseaux informatiques, ...

En effet, nous distinguons, selon la rationalité des individus deux classes différentes, la théorie des jeux classique et la théorie des jeux évolutionnaires. Chacune de ces classes est basée sur un ensemble différent d'hypothèses et cherche à atteindre un but bien particulier.
Ce présent chapitre sera consacré à la présentation de ces deux classes de jeux, leurs notions de base ainsi que leurs concepts de solution.

2.2 Description d'un jeu

Un jeu est une situation où des agents (les joueurs) sont conduits à faire des choix parmi un certain nombre d'actions possibles et dans un cadre défini à l'avance (les règles du jeu). Les résultats de ces choix constituent une issue du jeu à laquelle est associée un gain pour chacun des participants. Ces résultats ne dépendent pas de la décision d'un seul joueur, mais plutôt de celles de tous les autres avec la possibilité que le hasard intervienne.

2.2.1 Composantes d'un jeu

- **Joueurs :** un joueur est une personne physique, une société ou encore la nature .... Dans un jeu on peut distinguer un ensemble de I={1,..., N} joueurs ;
- **Stratégies :** chaque joueur i ∈ I a un ensemble de choix possibles S_i appelés stratégies.
  On note s_{-i}=(s_1, ..., s_{i-1}, s_{i+1}, ..., s_N) l'ensemble des stratégies de tous les joueurs, sauf la stratégie du joueur i ;
- **Utilité :** pour chaque joueur i correspond une fonction d'utilité u_i

$$u_i : S_1 \times ... \times S_N \rightarrow \mathbb{R} ,$$

  qui associe un gain pour chaque vecteur de stratégies $(s_1, s_2, \ldots, s_N)$.

Ainsi, un jeu est décrit comme suit :

$$< I, \{S_i\}_{i\in I}, \{u_i\}_{i\in I} > . \tag{2.1}$$

- **Rationalité :** on dit qu'un joueur possède un comportement rationnel s'il est conscient des alternatives et choisit délibérément la stratégie qui lui est le plus favorable parmi son ensemble d'actions.

Si le hasard ou la chance est présente dans le jeu et peut affecter le déroulement du jeu, il est courant de la transformer en un joueur fictif nommé "chance" ou "nature".

2.2.2 Notion de stratégie

La notion de stratégie constitue un concept central en théorie des jeux. Une stratégie est la spécification complète du comportement d'un joueur dans n'importe quelle situation, nous trouvons alors les stratégies pures et mixtes.
Notons par :

- S_i : l'ensemble des stratégies pures du joueur i ;
- Δ_i : l'ensemble des stratégies mixtes du joueur i.

Définition 2.2.1 *Une stratégie pure du joueur i est un plan d'actions qui prescrit une action de ce joueur à chaque fois qu'il est susceptible de jouer.*

Définition 2.2.2 *Une stratégie mixte est une stratégie où le joueur choisit au hasard le coup qu'il joue parmi les coups possibles. Cela revient à attribuer une certaine distribution de probabilités sur l'ensemble des stratégies pures du jeu. Ainsi, l'ensemble des stratégies mixtes d'un joueur i est défini comme suit :*

$$\Delta_i = \{\alpha = (\alpha_1, \ldots, \alpha_{n_i}) \in \mathbb{R}^{n_i} / \alpha_j \in [0,1], \forall j = \overline{1, n_i}, \sum_{j=1}^{n_i} \alpha_j = 1\},$$

où :

- n_i *est le nombre de stratégie pures du joueur i.*
- α_i *est la probabilité que la stratégie* s_i *soit jouée.*

2.3 Typologie des jeux

Les applications variées de la théorie des jeux sont dues aux différentes typologies dont elle dispose et que nous définirons ci-dessous.

2.3.1 Jeux coopératifs / non coopératifs

Un jeu est dit coopératif si les joueurs peuvent se grouper dans des coalitions où le choix de leurs stratégies est décidé en commun, afin d'améliorer les gains des joueurs coalisés [16]. Quant aux jeux non coopératifs, ils correspondent à des situations dans lesquelles chaque joueur décide de ses actions individuellement sans consulter les autres joueurs et n'offrent pas la possibilité d'une coopération formelle ou liante. Les jeux non coopératifs peuvent être divisés en deux catégories : jeux statiques et jeux dynamiques.

2.3.2 Jeux statiques / dynamiques

On dit qu'un jeu est statique ou simultané lorsque les joueurs choisissent leurs actions simultanément, puis reçoivent leurs gains respectifs. Et on dit qu'un jeu est dynamique, ou alterné, lorsque les joueurs choisissent leurs actions alternativement.

2.3.3 Jeux à information parfaite / imparfaite

Si au moment de prendre une décision, les joueurs sont au courant de tous les choix passés de leurs rivaux, alors on est en présence d'un jeu à information parfaite. Si au moins un des joueurs ne connaît pas, à un moment du déroulement du jeu, ce qu'a joué un des autres joueurs, alors on est dans un jeu à information imparfaite.

2.3.4 Jeux à information complète / incomplète

Un jeu est dit à information complète si chacun des joueurs connaît la structure (régles) du jeu. Le jeu est dit à information incomplète si, au moins, un des joueurs ne connaît pas entièrement la structure du jeu.

2.3.5 Jeux symétriques

Un jeu à deux joueurs est dit symétrique, si les joueurs ont accès au même ensemble de stratégies et la même fonction d'utilité, c'est à dire :

$$\begin{cases} & S_1 \ = \ S_2 \ = \ S, \\ et & \\ & \forall (s_1, \ s_2) \ \in \ S, \ u_1(s_1, \ s_2) \ = \ u_2(s_2, \ s_1) \end{cases}$$

2.3.6 Jeux à somme nulle / non nulle

Ce sont des jeux dans lesquels les intérêts sont parfaitement antagonistes. Ce qui est gagné par un joueur est perdu par l'autre. La somme des fonctions d'utilité est

donc nulle. Mais avec ce type de jeux ce n'est pas toujours évident de représenter la réalité d'où la nécessité d'introduire les jeux à somme non nulle.

2.3.7 Jeux finis

Le jeu (2.1) est dit fini, si tous les ensembles de stratégies des joueurs sont finis, c'est à dire

$$< I,\ \{S_i\}_{i\in I},\ \{u_i\}_{i\in I} >, \tag{2.2}$$

cardinal de S_i= $|S_i| < \infty$, $\forall$ i.

2.4 Formes des jeux

Un jeu peut être représenté sous deux formes : la forme stratégique et la forme extensive.

2.4.1 Jeux sous forme stratégique (normale)

La forme stratégique est adaptée pour les jeux à décisions simultanées, elle est pratique pour représenter un jeu ainsi que les stratégies de chaque joueur.

Exemple 2.4.1 *Le dilemme du prisonnier*

Deux individus sont arrêtés par la police suite à un vol à main armée et ils sont enfermés dans deux cellules séparées sans possibilité de communication. Chaque individu est interrogé séparément et il a le choix entre nier d'avoir commis le vol (stratégie N) ou dénoncer son complice comme seul responsable (stratégie D).
Nous avons donc un jeu non-coopératif avec :

1. *Joueurs : prisonnier1, prisonnier2 ;*
2. *Stratégies :* $S_1 = S_2$=*{N,D} ;*
3. *Gains : les gains des individus représentent le nombre d'années de prisons.*
   $u_1(N,N) = u_2(N,N) = -1$,
   $u_1(N,D) = u_2(D,N) = -10$,
   $u_1(D,N) = u_2(N,D) = 0$,
   $u_1(D,D) = u_2(D,D) = -8$.

Le tableau suivant représente la forme stratégique de ce jeu :

2.4.2 Jeux sous forme dynamique (extensive)

La forme extensive constitue une autre manière de décrire un jeu, elle se modélise par un arbre appelé arbre de Kuhn. Nous allons maintenant définir ses composantes :

- Un **nœud** est un point de jeu en lequel un joueur entreprend une action ou encore en lequel le jeu se termine ;

	prisonnier2	
	N	D
N	(-1,-1)	(-10, 0)
prisonnier1		
D	(0,-10)	(-8, -8)

TABLE 2.1 – Dilemme du prisonnier représenté sous forme stratégique

- Un **successeur** du nœud X est un nœud auquel on peut accéder aprés que X ait été atteint ;
- Un **prédéceseur** du nœud X est un nœud qui doit être atteint avant que X ne puisse lui-même être atteint ;
- Un **nœud initial** est un nœud sans prédécesseur ;
- Un **nœud terminal** est un nœud sans successeur et on lui associe un vecteur de nombres représentant les gains de chacun des joueurs ;
- Une **branche** est une action appartenant à l'ensemble d'actions d'un joueur en un nœud particulier ;
- Un **chemin** est une suite de nœuds et de branches partants du nœud initial au nœud terminal.

La figure suivante représente un jeu sous sa forme extensive :

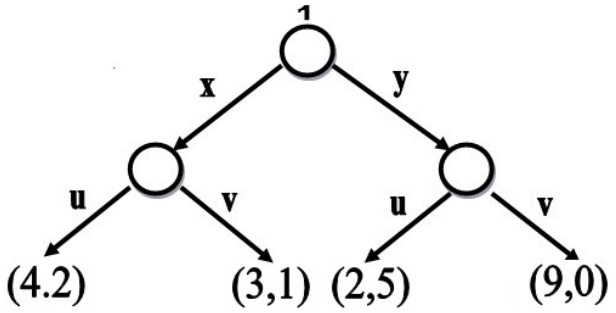


FIGURE 2.1 – Jeu sous forme extensive

2.5 Concepts de solution

L'analyse d'un jeu permet de prédire l'équilibre qui émergera du comportement des joueurs. Un équilibre est un état, ou une situation, dans lequel aucun joueur ne souhaite modifier son comportement une fois connu le comportement des autres joueurs.

2.5.1 Equilibre de Nash

L'équilibre de Nash doit son nom au mathématicien et économiste américain John F. Nash, qui a introduit ce concept en 1950. Dans ce qui suit, nous allons

définir l'équilibre de Nash en stratégies pures et mixtes.

Définition 2.5.1 *Un équilibre $s^* = (s_1^*, \ldots, s_N^*)$ est un équilibre de Nash en stratégies pures dans le jeu (2.2) si aucun joueur n'a intérêt à changer sa stratégie s_i^* au moment ou les autres joueurs continuent à jouer la stratégie s_{-i}^*, c'est à dire [26] :*

$$u_i(s_i^*, s_{-i}^*) \geq u_i(s_i, s_{-i}^*), \quad \forall s_i \quad \in S_i, \quad \forall i = 1, \ldots, N. \tag{2.3}$$

Définition 2.5.2 *Une situation $\alpha^* = (\alpha_1^*, \ldots, \alpha_N^*) \in \Delta = \Pi_{i=1}^N \Delta_i$ est un équilibre de Nash en stratégies mixtes dans le jeu (2.2) ssi :*

$$u_i(\alpha_i^*, \alpha_{-i}^*) \geq u_i(\beta_i, \alpha_{-i}^*), \quad \forall \beta_i \quad \in \Delta_i, \quad \forall i = 1, \ldots, N. \tag{2.4}$$

Proposition 2.5.1 *Tout équilibre de Nash en stratégies pures est aussi un équilibre de Nash en stratégies mixtes.*

Théorème 2.5.1 *Tout jeu fini admet au moins un équilibre de Nash en stratégies mixtes [17].*

Définition 2.5.3 *Une situation $\alpha^* = (\alpha_1^*, \alpha_2^*, \cdots, \alpha_N^*) \in \Delta = \prod_{i=1}^N \Delta_i$, est un équilibre de Nash strict dans le jeu (2.2), si on a :*

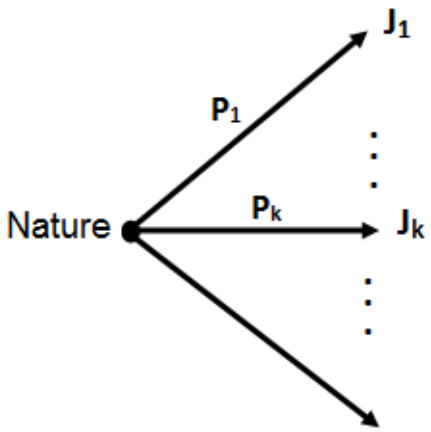
$$u_i(\alpha_i^*, \alpha_{-i}^*) > u_i(\beta_i, \alpha_{-i}^*), \quad \forall \beta_i \quad \in \Delta_i, \quad \forall i = 1, \ldots, N. \tag{2.5}$$

2.6 Jeux Bayésiens

Un jeu Bayésien est une situation conflictuelle à information incomplète entre deux ou plusieurs joueurs. Harsanyi (1967-68) est le premier à développer l'outil nécessaire pour la modélisation de ce type de situation.

Considérons un jeu J opposant deux joueurs :

- Le joueur 1 connaît parfaitement les gains résultant de chaque paire de stratégies possibles, contrairement au joueur 2. Ce dernier ignore donc une composante essentielle du jeu. Chacun connaît néanmoins l'ensemble de stratégies, supposé fini, de chaque joueur ;
- Comme le joueur 2 ne connaît pas précisément les gains du jeu auquel il participe, il possède des croyances sur ces gains. Ces croyances sont représentées sous la forme d'un ensemble de fonctions de gains possibles et de probabilités attachées à chacune de ces fonctions ;
- Notons par J_k le jeu correspondant à la $k^{ème}$ fonction de gain et p_k la probabilité que le vrai jeu auquel le joueur est confronté soit J_k. Chacun de ces jeux a exactement la même structure, sinon le joueur 2 pourrait deviner à quel jeu il participe en observant cette structure ;
- Le joueur 2 sait que le joueur 1 connaît parfaitement le jeu.



La manière dont le joueur 2 perçoit le jeu peut être représentée, selon Harsanyi, par un coup initial d'un joueur fictif, la nature, qui choisit le jeu. Comme l'illustre la figure suivante :

D'abord, la nature tire au hasard le vrai jeu auquel doivent jouer les deux joueurs, ensuite le joueur 1 est informé de ce choix et le jeu débute. Le dé qu'utilise la nature possède autant de faces que le nombre de jeux possibles et ce dé est truqué de telle manière que la probabilité p_k que le jeu J_k soit tiré corresponde exactement aux croyances du joueur 2. Harsanyi propose donc de remplacer le jeu J à information incomplète par un jeu séquentiel à information complète (mais imparfaite),$J^{'}$, où la nature joue en premier.

Un joueur peut être de plusieurs types différents. Ainsi, chaque fois que l'incertitude porte sur une des caractéristiques d'un joueur, nous pouvons représenter les différents états de la Nature comme des types différents de ce joueur.

Définition 2.6.1 *[26]*
Un jeu Bayésien est décrit par les éléments suivants :

- *Un ensemble de N joueurs : I={1, . . . , N} ;*
- *Pour chaque joueur i, i∈ I :*
  - *Un ensemble de stratégies S_i, qui contient toutes les stratégies possibles de ce joueur et un ensemble T_i qui contient ses différents types possibles,*
  - *Une fonction de gain U_i :*

$$U_i : \ (\prod_{i\in I} S_i) \times (\prod_{i\in I} T_i) \to \mathbb{R}$$

$$(s_1, s_2, \ldots, s_N; t_1, \ldots, t_N) \to U_i(s;t). \tag{2.6}$$

  - *Une distribution de probabilités p_i qui donne ses croyances quant aux types des autres joueurs*

$$p_i : \ T \to [0,1]$$

$$(t_1, \ldots, t_N) \rightarrow p_i(t_{-i}|t_i).$$

qui résulte de la règle de Bayes quand elle peut être appliquée après révélation de son type au joueurs i

$$p_i(t_{-i}|t_i) = \frac{p(t_{-i}, t_i)}{p(t_i)} = \frac{p(t_{-i}, t_i)}{\sum_{t_{-i} \in T_i} p(t_{-i}, t_i)}. \tag{2.7}$$

Les stratégies d'un joueur dans ce jeu doivent être construites à partir de l'ensemble des stratégies du joueur et de son ensemble de types :

Définition 2.6.2 *[26]*
Une stratégie du joueur i n'est pas directement donnée dans la définition du jeu, mais elle est définie de manière à préciser une action pour chaque type $t_i \in T_i$ à partir de l'ensemble des stratégies S_i :

$$a_i : \ T_i \rightarrow S_i$$

$$t_i \rightarrow a_i(t_i) = s_i. \tag{2.8}$$

L'équilibre de Nash de ce type de jeu peut être défini comme suit.

Définition 2.6.3 *[26]*
Une situation $a^*=(a_1^*, \ldots, a_N^*)$ est un équilibre de Nash en stratégies pures d'un jeu bayésien si $a_i^*(t_i)$ est la solution du problème suivant pour chaque joueur i et pour chacun des types $t_i \in T_i$ de ce joueur ($\forall i$, $\forall t_i \in T_i$) :

$$\max_{s_i \in S_i} \sum_{t_{-i} \in T_{-i}} U_i(a_1^*(t_1), \ldots, a_{i-1}^*(t_{i-1}), s_i, a_{i+1}^*(t_{i+1}), \ldots, a_N^*(t_N); t) \times p_i(t_{-i}|t_i). \tag{2.9}$$

2.7 Jeux de signalisation de base

Les jeux de signalisation se réfèrent à une classe de jeux à deux joueurs à information incomplète dans laquelle un seul joueur est informé seulement.

2.7.1 Description du jeu

Un jeu de signalisation basique, dans sa forme la plus simple, est défini comme suit :

- **Joueurs :** il existe trois joueurs en tout
  1. Joueur 1 : est l'expéditeur, avec un ensemble d'informations privées ;
  2. Joueur 2 : est le récepteur dont l'ensemble d'information est connu de tous ;
  3. Joueur 0 : est un joueur fictif, la nature qui tire le type θ de l'expéditeur à partir d'un ensemble Θ de types.
- **Stratégies :** chacun des deux joueurs, 1 et 2, dispose d'un ensemble de stratégies S_1 et S_2 respectivement :
  – $S_1 = \{s_1^1, s_2^1, \ldots, s_{n_1}^1\}$;

– $S_2 = \{s_1^2,\ s_2^2, \ldots,\ s_{n_2}^2\}$,
où :
s_j^i : représente la $j^{ème}$ stratégie du $i^{ème}$ joueur et n_1, n_2 représentent le nombre total de stratégies des joueurs 1 et 2, respectivement.

L'espace des stratégies mixtes sont Δ_{n_1} et Δ_{n_2} avec les vecteurs de probabilités α^1 et α^2 respectivement. Le joueur 1 observe l'information concernant son type θ et choisit une action $s_i^1 \in S_1$. Le joueur 2, dont le type est connu de tous, observe s_i^1 et choisit une action $s_i^2 \in S_2$. Avant le début du jeu, le second joueur possède des croyances sur le type, θ, du joueur 1 à base desquelles il attribue une probabilité $p(\theta)$ pour chacun des types θ.
La stratégie du joueur 1 est une distribution de probabilité $\sigma_1(.|\theta)$ sur les actions s_i^1 pour chaque type θ, quant à la stratégie du joueur 2 est une distribution de probabilités $\sigma_2(.|s_i^1)$ sur les actions s_i^2 pour chaque stratégie s_i^1 du joueur 1.

- **Utilité :** après que les deux joueurs aient pris leurs actions dans leurs espaces de stratégies respectifs, les utilités sont attribuées selon ces deux actions ainsi que le type de l'expéditeur choisi par la nature. La fonction d'utilité de chacun des joueurs est définie de la manière suivante :

$$u_i : S_1 \times S_2 \times \Theta \to \mathbb{R}, \qquad i = 1, 2.$$

L'utilité du joueur 1 en choisissant la stratégie $\sigma_1(.|\theta)$ et que le joueur 2 joue $\sigma_2(.|s_i^1)$ est :

$$u_1(\sigma_1, \sigma_2, \theta) = \Sigma_{i=1}^{n_1}\Sigma_{j=1}^{n_2}\sigma_1(s_i^1|\theta)\sigma_2(s_j^2|s_i^1)u_1(s_i^1, s_j^2, \theta). \tag{2.10}$$

L'utilité du joueur 2 à la stratégie $\sigma_2(.|s_i^1)$, lorsque le joueur 1 joue $\sigma_1(.|\theta)$ est :

$$u_2(\sigma_1, \sigma_2, \theta) = \Sigma_\theta p(\theta)(\Sigma_{i=1}^{n_1}\Sigma_{j=1}^{n_2}\sigma_1(s_i^1|\theta)\sigma_2(s_j^2|s_i^1)u_2(s_i^1, s_j^2, \theta)). \tag{2.11}$$

Le récepteur met à jour ses croyances, $p(\theta)$, sur le type de l'expéditeur et base son choix d'action s_i^2 sur la distribution à posteriori [1] $\mu(.|s_i^1)$ sur Θ. L'équilibre Bayésien parfait, que nous définirons ci-dessous, dicte que l'action du joueur 1 notée $\sigma_1^*(.|\theta)$ dépend de son type. En connaissant $\sigma_1^*(.|\theta)$ et en observant s_i^1, le joueur 2 peut utiliser la règle de Bayes afin de mettre à jour $p(.)$ et $\mu(.|s_i^1)$.

2.7.2 Equilibre Bayésien parfait

Définition 2.7.1 *Un équilibre Bayésien parfait d'un jeu de signalisation est le profil de stratégie σ^* et les croyances à posteriori $\mu(.|a_i^1)$ qui vérifient les conditions suivantes [18] :*

(P_1) **:** $\forall\theta, \sigma_1^*(.|\theta) \in arg\max_{\alpha_1} u_1(\alpha_1, \sigma_2^*, \theta)$;

(P_2) **:** $\forall s_i^1 \in S^1, \sigma_2^*(.|s_i^1) \in arg\max_{\alpha_2} \Sigma_\theta \mu(\theta|s_i^1)u_2(s_i^1, \alpha_2, \theta)$;

1. Une distribution à posteriori est la nouvelle répartition calculée à partir d'une expérience en utilisant le théorème de Bayes.

(B) : $\mu(\theta|s_i^1) = \frac{p(\theta)\sigma_1^*(s_i^1|\theta)}{\Sigma_{\theta^1\in\Theta}p(\theta^1)\sigma_1^*(s_i^1|\theta^1)}$, *si* $\Sigma_{\theta^1\in\Theta}p(\theta^1)\sigma_1^*(s_i^1|\theta^1) > 0$.

Dans le cas où $\Sigma_{\theta^1\in\Theta}\ p(\theta^1)\sigma_1^*(s_i^1|\theta^1) = 0$, $\mu(.|s_i^1)$ *est une distribution de probabilités quelconque sur* Θ.
(P_1*) et (*P_2*) sont les conditions de perfection et (B) correspond à l'application de la règle de Bayes.*

(P_1) signifie que le joueur 1 prend en compte l'effet de s_i^1 sur l'action du joueur 2.
(P_2) signifie que le joueur 2 réagit de manière optimale à l'action du joueur 1 avec ses croyances à posteriori sur θ.

2.8 Jeux évolutionnaires

La théorie des jeux évolutionnaires est une nouvelle classe de jeux qui a été utilisée en premier par les biologistes pour comprendre le comportement animal où ils ont constaté que ce dernier est peu rationnel au sens décrit par la théorie classique des jeux. Les jeux évolutionnaires, introduits par John Maynard Smith [21], cherchent à étudier les solutions auxquelles peuvent conduire les interactions en l'absence d'une rationalité forte. Il considère alors que le jeu est joué un grand nombre de fois entre des individus possédant une rationalité limitée et très peu d'information sur le jeu. Les joueurs sont tirés de manière aléatoire à partir d'une grande population [26]. Cependant, l'interaction entre les différentes populations ou entre des comportements différents se fait à travers beaucoup d'interactions locales entre un petit nombre d'individus, où chacun cherche à améliorer, non pas son gain personnel, mais le gain total de la population dont il fait partie.

Cette catégorie de jeux cherche à étudier la dynamique de l'évolution en se basant sur deux concepts fondamentaux, les stratégies évolutionnairement stables et le réplicateur dynamique, mais avant d'arriver à cela, définissons d'abord les notions liées aux jeux évolutionnaires.

2.8.1 Définitions et concepts

Un ensemble de définitions et de concepts est nécessaire afin d'introduire et d'immerger dans la théorie des jeux évolutionnaires.

Définition 2.8.1 *Une* ***population*** *est un ensemble d'individus qui coexistent dans le même environnement.*
Un individu peut être une personne, un animal, une entreprise, une machine (ordinateur), $\cdots$

Définition 2.8.2 *Un* ***comportement*** *particulier, ou une suite de comportements, que l'individu adopte est appelé stratégie.*

Définition 2.8.3 *Un individu est dit* ***mutant****, s'il change son comportement (sa stratégie) au cours du temps par rapport à son comportement initial.*

Définition 2.8.4 *Une **stratégie mutante** est celle adoptée par un mutant.*

Définition 2.8.5 *Une **stratégie originelle** est une stratégie qui n'est pas une stratégie mutante.*

Définition 2.8.6 *Une stratégie α est dite **envahie** par une stratégie β, si les individus jouant la stratégie β obtiennent des gains plus élevés que les autres individus de la population jouant la stratégie α.*

Définition 2.8.7 ***Profil de la population***
On considère une population infinie d'individus pouvant utiliser un ensemble de stratégies pures S.
Le profil de population est un vecteur p dont chaque composante donne une probabilité avec laquelle chaque stratégie est jouée dans la population.

Exemple 2.8.1 *Nous considérons un ensemble de stratégies $S = \{s_1, s_2, s_3\}$. Si la population est constituée de 40% utilisant la stratégie mixte $(\frac{1}{2}, 0, \frac{1}{2})$ et 60% utilisant $(\frac{1}{4}, \frac{3}{4}, 0)$, alors le profil de cette population serait :*

$$p = \frac{4}{10}(\frac{1}{2}, 0, \frac{1}{2}) + \frac{6}{10}(\frac{7}{10}, \frac{3}{10}, 0).$$

Cette notion est très utilisée par un autre type de jeux appelé : jeux de population.

Définition 2.8.8 *Un état de la population est dit [25] :*
- ***Monomorphique**, si chaque individu de la population utilise la même stratégie ;*
- ***Polymorphe**, si au moins deux individus de la population utilisent deux stratégies différentes.*

Définition 2.8.9 *Une stratégie α est **stable**, si le gain des individus qui adoptent cette stratégie est supérieur au gain de tout autre mutant. S'il existe une stratégie mutante qui peut envahir la stratégie α, alors celle-ci est dite instable.*

2.8.2 Stratégie Evolutionnairement Stable (ESS)

Une stratégie évolutionnairement stable est un concept clé dans la théorie des jeux évolutionnaires. Elle se définit comme étant une stratégie qui résiste aux pressions évolutionnaires exercées par l'environnement : une population jouant une telle stratégie ne peut être envahie par aucune autre stratégie mutante.
J. Maynard Smith a donné des conditions mathématiques pour qu'une stratégie soit évolutionnairement stable et les hypothèses essentielles qui ont été utilisées dans son modèle consistaient à considérer :

- Une population infinie ;
- Les combats se font par paire d'individus ;
- Chaque paire de concurrents est considérée comme étant deux adversaires qui ont les mêmes aptitudes ; (c'est-à-dire un combat symétrique) ;

• Chaque individu de la population dispose d'un même ensemble fini de stratégies.

Ceci permet de décrire le jeu entre deux individus par une matrice des gains A, ce qui donne des jeux symétriques matriciels à deux joueurs. Chaque élément a_{ij} de la matrice représente le gain de l'individu quand il choisit de jouer sa $i^{ième}$ stratégie pure et son adversaire sa $j^{ième}$ stratégie pure. Par conséquent, nous sommes amenés à considérer des jeux finis à deux joueurs avec des matrices des gains $A_1 = A = (a_{ij})_{i;j=\overline{1;n}}$ et $A_2 = A^T$ avec n : le nombre de stratégies de chacun des joueurs.

L'ensemble des stratégies mixtes associé à chaque individu est défini par le n-simplexe :

$$\Delta = \{\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)/\alpha_i \in [0,1], \forall i = \overline{1,n}, \sum_{i=1}^{n} \alpha_i = 1\}.$$

Comme l'évolution de la population est considérée comme la résultante de confrontation d'individus (de cette population) par paire et donc connue des jeux matriciels, alors si ces jeux sont traités en stratégies mixtes, le gain espéré pour chaque individu, s'il adopte une stratégie mixte $\alpha \in \Delta$ et son adversaire une stratégie mixte $\beta \in \Delta$, serait :

$$u(\alpha, \beta) = \alpha^T A \beta = \Sigma_{i=1}^{n} \Sigma_{j=1}^{n} a_{ij} \alpha_i \beta_j.$$

Ainsi, l'étude de l'évolution de la population se ramène à l'étude du jeu symétrique à deux joueurs en stratégies mixtes

$$< \mathcal{A}, \mathcal{B}, u_1, u_2 >, \tag{2.12}$$

où $\mathcal{A} = \mathcal{B} = \Delta$, et

$$\begin{cases} u_1(\alpha, \beta) = u(\alpha, \beta) = \alpha^T A_1 \beta = \alpha^T A \beta, \\ u_2(\alpha, \beta) = \alpha^T A_2 \beta = \beta^T A_2^T \alpha = \beta^T A \alpha = u(\alpha, \beta). \end{cases}$$

Le jeu (2.12) est entièrement caractérisé par :

$$< \Delta, u > . \tag{2.13}$$

2.6.2.1 Définition d'une ESS

Intuitivement, le concept de stratégie évolutionnairement stable peut être interprété de la manière suivante. Supposons qu'on a une population infinie d'individus et ces derniers se rencontrent aléatoirement par paire pour jouer un jeu à deux joueurs symétrique. Supposons aussi qu'à l'état initial, l'ensemble des individus de cette population adopte une même stratégie α (pure ou mixte).
Dans ce cas, on note par u(α, α) le gain d'un individu dans le jeu symétrique à deux joueurs.
Supposons maintenant qu'une proportion ε d'individus de cette population adopte une autre stratégie β (une stratégie mutante), tandis que le reste des individus de la population maintient la stratégie α.

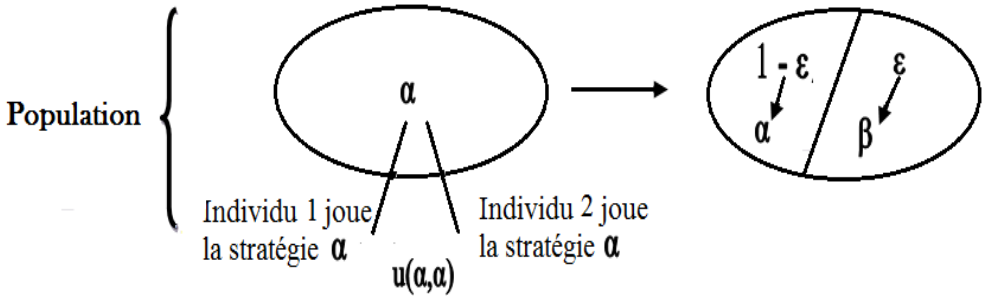


FIGURE 2.2 – Changement de stratégie dans la population.

Mettons nous à la place d'un individu jouant une stratégie α. Alors la probabilité d'être apparié à un joueur jouant une stratégie α (pour jouer le jeu symétrique à deux joueurs) sera de (1-ε) et la probabilité d'être apparié à un joueur jouant une stratégie β sera ε. Alors l'espérance de gain du joueur jouant une stratégie α sera :

$$\begin{aligned}(1-\varepsilon)u(\alpha,\alpha)+\varepsilon u(\alpha,\beta) &= (1-\varepsilon)\alpha^T A\alpha+\varepsilon\alpha^T A\beta,\\ &= \alpha^T A((1-\varepsilon)\alpha+\varepsilon\beta),\\ &= u(\alpha,(1-\varepsilon)\alpha+\varepsilon\beta).\end{aligned}$$

Par contre, l'espérance de gain du joueur jouant une stratégie β sera :

$$\begin{aligned}(1-\varepsilon)u(\beta,\alpha)+\varepsilon u(\beta,\beta) &= (1-\varepsilon)\beta^T A\alpha+\varepsilon\beta^T A\beta,\\ &= \beta^T A((1-\varepsilon)\alpha+\varepsilon\beta),\\ &= u(\beta,(1-\varepsilon)\alpha+\varepsilon\beta).\end{aligned}$$

La stratégie α sera gagnante face à la stratégie mutante β, si :

$$u(\alpha,(1-\varepsilon)\alpha+\varepsilon\beta)>u(\beta,(1-\varepsilon)\alpha+\varepsilon\beta).$$

Cette intuition fournit la définition d'une ESS.

Définition 2.8.10 *(Taylor et Joncker 1978)*
Une stratégie $\alpha\in\Delta$ est une Stratégie évolutionnairement stable (ESS), si :
$\forall\beta\in\Delta,\exists\bar{\varepsilon}=\bar{\varepsilon}(\beta)\in(0,1),\forall\varepsilon\in(0,\bar{\varepsilon}),$

$$u(\alpha,(1-\varepsilon)\alpha+\varepsilon\beta)>u(\beta,(1-\varepsilon)\alpha+\varepsilon\beta). \tag{2.14}$$

ou bien,

$$\alpha^T A((1-\varepsilon)\alpha+\varepsilon\beta)>\beta^T A((1-\varepsilon)\alpha+\varepsilon\beta).$$

Proposition 2.8.1
Une stratégie $\alpha\in\Delta$ est une ESS, si et seulement si :
i) $\forall\beta\in\Delta,\ \alpha^T A\alpha\geq\beta^T A\alpha.$

ii) $\forall \beta \in \Delta, \beta \neq \alpha$, *si*

$$\alpha^T A\alpha = \beta^T A\alpha \Rightarrow \alpha^T A\beta > \beta^T A\beta.$$

Cette proposition correspond à la formalisation initiale des stratégies évolutionnairement stables donnée par J. Maynard Smith. Elle est considérée par certains auteurs comme étant une définition équivalente d'une stratégie évolutionnairement stable. Cette proposition représente une condition nécessaire et suffisante pour l'existence d'une stratégie évolutionnairement stable.

2.6.2.2 Relation entre l'équilibre de Nash et les ESS

Les deux propositions suivantes formulent la relation existant entre l'équilibre de Nash et les ESS.

Proposition 2.8.2

1. *Si une stratégie mixte $\alpha \in \Delta$ est une ESS dans le jeu (2.13), alors α est un équilibre de Nash dans le même jeu ;*
2. *Si une stratégie $\alpha \in \Delta$ est un équilibre de Nash strict, alors α est une ESS.*

2.6.2.3 ESS dans le jeu du Faucon et de la Colombe (Hawk and Dove)

Dans cette section, nous donnerons un exemple bien connu dans la théorie des jeux évolutionnaires pour illustrer le concept de l'ESS défini précédemment.

Exemple 2.8.2 *(Jeu du faucon-colombe)*

On dispose d'une très grande population d'animaux (individus) d'une même espèce qui se disputent un territoire. Ces individus se rencontrent aléatoirement par paire (les combats se font par paire) pour jouer un jeu symétrique à deux joueurs. Dans la réalité, les animaux changent souvent leur comportement d'une manière très complexe.
Supposons que dans le jeu symétrique qui l'oppose à un autre individu (joueur) de la population, un individu peut adopter un comportement pacifique : on dira qu'il suit une stratégie du type Colombe (D) ou un comportement agressif : on dira qu'il suit une stratégie de type Faucon (H). Donc, tous les individus ont le même ensemble de stratégies $X = \{H;D\}$.

- ***H*** *: représente la stratégie faucon (Hawk), qui consiste à combattre à fond jusqu'à ce qu'il gagne (en blessant l'autre ou en le faisant fuir) ou qu'il soit lui même blessé ;*
- ***D*** *: représente la stratégie colombe (Dove), qui consiste à combattre conventionnellement et s'enfuir dès que ça devient dangereux, avant d'être blessé.*

Deux individus (animaux) pris aléatoirement de la population se disputent un territoire dont la valeur est V. A la fin de chaque affrontement, un individu jouant la stratégie α et son adversaire jouant la stratégie β reçoit un gain $u(\alpha, \beta)$. Ce gain

est déterminé par les trois facteurs suivants : le gain (gain du territoire), la perte (perte du territoire) si on est blessé et la perte résultant d'un conflit prolongé évaluée à C.
L'ensemble des stratégies est donné par X = {H,D}. Les gains de toutes les configurations qui peuvent se présenter sont comme suit :

- *Lorsque les deux individus adoptent tous les deux une stratégie D, alors ils partagent pacifiquement le territoire et ne subissent aucune perte, ce qui donnera un gain :*
$$u(D, D) = \frac{V}{2}.$$
- *Lorsque le joueur adopte la stratégie agressive H et son adversaire la stratégie pacifique D, alors il obtiendrait la totalité du territoire :*
$$u(H, D) = V.$$
- *Lorsque le joueur adopte la stratégie pacifique D et son adversaire la stratégie agressive H, alors il ne gagnerait rien et ne subirait pas de perte :*
$$u(D, H) = 0.$$
- *Lorsque le joueur adopte la stratégie agressive et son adversaire adopte le même comportement, alors il aurait une chance sur deux pour avoir le territoire et de subir des pertes dues aux blessures :*
$$u(H, H) = \frac{V}{2} - \frac{C}{2}.$$

Ainsi, la matrice des gains associée est :

$$A = \begin{array}{c} \\ H \\ D \end{array} \begin{array}{c} \begin{array}{cc} H & D \end{array} \\ \begin{pmatrix} \frac{V-C}{2} & V \\ 0 & \frac{V}{2} \end{pmatrix} \end{array}$$

Matrice 2.1 Forme stratégique du jeu Faucon-Colombe.

L'étude du jeu, de manière globale, commence par considérer les stratégies pures. Trois configurations sont possibles : V < C, V = C et V > C.

Recherche des ESS

Si V > C :
$$u(H, H) = \frac{V - C}{2} > u(D, H) = 0.$$
D'après la définition 2.5.3, (H, H) est un équilibre de Nash strict et donc une ESS d'après la proposition 2.6.2.

Si V = C :
$$0 = \frac{V - C}{2} = u(H, H) = u(D, H) = 0.$$
Pareil que dans le premier cas, la stratégie H est l'ESS unique du jeu.

Si V < C :

$$u(H,H) = \frac{V-C}{2} < u(D,H) = 0.$$

Dans ce cas, nous voyons que $\frac{V-C}{2} < 0$ et d'autre part $\frac{V}{2} < V$, donc H et D ne sont pas des ESS en stratégies pures.

Nous voyons bien que dans cet exemple que si $V < C$, le jeu n'admet pas de stratégies évolutionnairement stables en stratégies pures.

2.8.3 Réplicateur dynamique

La théorie des jeux évolutionnaires permet de modéliser explicitement la dynamique présente dans les interactions entre les joueurs. Cette branche de la théorie des jeux analyse l'évolution de la fréquence des stratégies (comportements) au sein d'une population d'agents interagissant stratégiquement et dotés d'une rationalité nulle ou limitée.

Les processus évolutionnaires comportent deux mécanismes de base : la mutation et la sélection des stratégies. La stabilité évolutionnaire souligne le rôle des stratégies mutantes dans la mesure où l'on étudie la stabilité des solutions du jeu face aux mutations dans la population des stratégies [26]. Quant au modèle de réplication dynamique développé par Taylor et Jonker (1978)[11], il décrit un processus de sélection spécifiant comment une population est associée avec différentes stratégies pures dans un jeu qui évolue dans le temps.

1.6.3.1 Modèle

Le modèle des replicateurs met en succès une reproduction asexuée d'individus d'une population de différents types $\{C_1, \ldots, C_n\}$ adoptant chacun un comportement. Le coportement d'un individu de la population de type C_i est décrit par une stratégie mixte ou pure.

Considérons une large population finie d'individus, jouant chacun une stratégie pure $s_i \in \mathrm{S} = \{s_1, s_2, \ldots, s_n\}$, qui sont tirés au hasard par paires pour jouer un jeu fini symétrique à deux joueurs.
Soit $m_i(t)$ le nombre d'individus qui jouent la stratégie pure $s_i \in \mathrm{S}$ au temps t. Ainsi, m(t) qui est le nombre total d'individus dans la population est :

$$m(t) \;=\; \sum_{i=1}^{n} m_i(t).$$

Soit $p_i(t) = \frac{m_i(t)}{m(t)}$ la proportion d'individus de la population jouant la stratégie pure $s_i \in \mathrm{S}$ au temps t. L'état de la population est alors :

$$p(t) \;=\; (p_1(t),\; p_2(t),\; \ldots,\; p_n(t)).$$

Remarque 2.8.1 *L'état p(t) de la population est formellement identique à une stratégie mixte du jeu à deux joueurs à l'état initial.*

Modèle discret

Dans ce modèle, à chaque période de temps, les individus sont tirés au hasard par paires pour jouer exactement une instance du jeu. Ces individus prennent connaissance de leurs gains respectifs et des gains des autres à la fin de la période, ceci leur permet d'utiliser ces informations de manière à réviser le choix de leurs stratégies.
A la fin de la période, phase de reproduction : chaque individu est remplacé par un nombre d'individus jouant la même stratégie proportionnel au gain reçu lors de la confrontation.
Le nombre moyen de remplacements pour un individu jouant la stratégie s_i est proportionnel au gain moyen procuré par la stratégie s_i, soit [26] :

$$U(s_i,\ p)\ =\ EU(s_i,\ p)\ =\ \sum_{j=1}^{n} U(s_i,\ s_j)p_j\ =\ \sum_{j=1}^{n} a_{ij}p_j\ =\ (Ap)_i, \tag{2.15}$$

où : A=$(a_{ij})_{i,j=\overline{1,n}}$.

Par conséquent, pour un individu jouant la stratégie pure s_i, il n'y a pas de différence entre une rencontre avec un autre individu tiré aléatoirement de la population jouant la stratégie p et jouer face à un vrai joueur jouant la stratégie mixte p.

Le gain moyen dans la population, et donc le gain espéré, d'un individu participant à une rencontre aléatoire dans la population jouant la stratégie mixte p est donné par :

$$U(p,\ p)\ =\ \sum_{i=1}^{n} p_i U(s_i,\ p)\ =\ p^T Ap. \tag{2.16}$$

Le gain U(p, p) est équivalent au gain espéré de la stratégie mixte p jouant contre elle même.

La proportion d'individus de la population, adoptant la stratégie s_i à la période t+1 est donnée par :

$$p_i(t+1)\ =\ p_i(t)\frac{U(s_i,\ p(t))}{U(p(t),\ p(t))}\ =\ p_i(t)\frac{(Ap(t))_i}{(p(t))^T Ap(t)}. \tag{2.17}$$

Ainsi, le processus caractérisant l'évolution de la population est :

$$\Delta p_i(t)\ =\ p_i(t{+}1) - p_i(t)\ =\ p_i(t)[\frac{(Ap(t))_i}{p(t)^T Ap(t)} - 1]\ =\ p_i(t)[\frac{(Ap(t))_i\ -\ p(t)^T Ap(t)}{p(t)^T Ap(t)}].$$

En abandonnant l'indice relatif au temps, la dernière expression peut se réécrire :

$$\Rightarrow\quad \Delta p_i(t)\ =\ p_i(t+1)\ -\ p_i(t)\ =\ p_i\frac{(Ap)_i\ -\ p^T Ap}{p^T Ap}. \tag{2.18}$$

Le système d'équations aux différences Δp décrit le processus de réplication à temps discret.

Modèle continu

Ce modèle est appliqué quand les périodes de temps deviennent courtes. Dès lors, l'équation aux différences Δp_i peut être approximée par l'équation différentielle suivante :

$$\frac{dp_i}{dt} \;=\; \dot{p_i} \;=\; p_i \frac{(Ap)_i \;-\; p^T Ap}{p^T Ap}. \tag{2.19}$$

Ces équations ont des propriétés équivalentes à celles de l'équation simplifiée :

$$\dot{p_i} = p_i[(Ap)_i \;-\; p^T Ap]. \tag{2.20}$$

Ce dernier système décrit le processus de réplication en temps continu. Il donne le pourcentage d'individus jouant la stratégie s_i nouvellement dénombrée au sein de la population à la prochaine période et il dépend de la valeur initiale $p_i(t_0)$.

Remarque 2.8.2 *Des stratégies non jouées au début du jeu (à l'instant initial) ne peuvent pas apparaître dans la dynamique de réplication.*

En posant :

$$p(t) \;=\; (p_1(t),\; \ldots,\; p_n(t)),$$

L'équation de la dynamique de réplication (2.20) peut s'écrire sous la forme :

$$\frac{dp}{dt} = \dot{p} = f(p), \tag{2.21}$$

où :

$$f(p) = (f_1(p),\; \ldots,\; f_n(p)),\; f_i(p) = p_i[(Ap)_i \;-\; p^T Ap].$$

et $p(0) = (p_1(0);\; p_2(0);\; \ldots;\; p_n(0))$ est l'état de la population à l'instant initial. On note par $p(p(0);\; t)$, l'application $p : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$, telle que $p(p(0); t)$ est l'état de la population à l'instant t en partant de la condition initiale $p(0)$ en t = 0.

Dans plusieurs cas, il est impossible de trouver une solution explicite pour l'équation du réplicateur dynamique (2.21), mais il peut être possible de trouver son comportement asymptotique.

Etat stationnaire et stabilité

Dans le réplicateur dynamique nous nous intéressons uniquement aux résultats à long terme. Le comportement asymptotique du système pourrait parfois être déterminé sans trouver d'abord la solution explicite.

Définition 2.8.11 *Etat (point) stationnaire*
$p^* \in \Delta$ est un point stationnaire ou un point d'équilibre pour l'équation différentielle (2.21) définie sur $\Delta \subseteq \mathbb{R}^n$, si :

$$f(p^*) = 0. \tag{2.22}$$

L'une des propriétés fondamentales d'un état stationnaire est que si le système débute en cet état il y reste pour l'éternité. Certains états stationnaires ne peuvent être atteints, si le système ne débute pas en eux. D'autres attireront les points de départ, ainsi le système va se rapprocher de ces états avec l'augmentation de t. Ceci est le concept des états stationnaires stables et instables

Définition 2.8.12 *Un état stationnaire est dit stable s'il attire des points près de lui, ceci signifie que si le système s'éloigne un peu de cet état il va y revenir, dans le cas contraire il est dit instable.*

Définition 2.8.13
$p^* \in \Delta$ est un point stable, s'il est un point stationnaire de l'équation différentielle (2.21) et pour tout voisinage $\mathcal{V}$ de p^*, il existe un voisinage $\mathcal{U} \subset \mathcal{V}$ tel que si $p(0) \in \mathcal{U}$ alors $p(p(0);\ t) \in \mathcal{V}$ pour tout $t > 0$.

Habituellement, les états stationnaires de l'équation du réplicateur dynamique sont d'un grand intérêt car on ne s'intéresse pas à ce qui va se passer à un instant déterminé t mais plutôt à ce qui va se passer à long terme, là où le système atteindra ou sera proche d'un état stationnaire.

2.9 Conclusion

La première partie de ce chapitre a été consacrée à la présentation et l'introduction des notions de base de la théorie des jeux classique. L'intérêt principal de cette théorie est d'étudier les différentes situations de conflit entre les individus en prenant comme hypothèse leur rationalité. L'étude de ces jeux se base sur la notion d'équilibre et en particulier d'équilibre de Nash qui définit une situation de non regret pour les différents joueurs.
La seconde partie a été dédiée à la théorie des jeux évolutionnaires qui constitue aujourd'hui un pilier essentiel de la théorie des jeux. L'idée principale sous-jacente aux jeux évolutionnaires est que ce n'est plus la rationalité de chaque individu qui le pousse à adapter son comportement aux stratégies de ses adversaires, mais une évolution propre à l'ensemble de la population à laquelle il appartient. L'analyse de ces jeux revient à étudier l'ESS et les états stables d'une population.

Le chapitre qui suit fera l'objet des travaux déjà réalisés sur la modélisation des problèmes de sécurité des réseaux ad hoc par la théorie des jeux.

3
Etat de l'art

3.1 Introduction

Les méthodes de sécurité informatique sont en plein développement et la difficulté qu'elles rencontrent est de suivre le rythme changeant de leur environnement et de saisir l'essence même du problème.

La théorie des jeux fournit des outils pour étudier l'interaction entre des joueurs dans une société. Pour cela, elle a été suggérée pour modéliser l'interaction entre un attaquant et un IDS dans différents types de réseau.

Dans ce chapitre, nous allons présenter quelques modèles de la littérature théorique sur l'insertion de la théorie des jeux dans les problèmes de sécurité des réseaux ad hoc.

3.2 Jeux de signalisation dans la sécurité des réseaux ad hoc

Les réseaux ad hoc, comme tout autre type de réseaux, peuvent être des cibles de maintes attaques qui peuvent causer des dommages et ainsi dégrader leurs performances. Pour contrer ces attaques, tous les nœuds composant ce réseau devront être équipés d'un IDS. Ainsi, une situation d'interaction se crée entre l'IDS, dont le but est de protéger le réseau, et l'attaquant qui cherche à compromettre la sécurité de ce réseau.

Nous trouvons dans [18], la modélisation de cette interaction dans un réseau ad hoc sous forme d'un jeu de signalisation de base qui est un jeu séquentiel non-coopératif à informations incomplètes. Comme nous le savons, un jeu non coopératif

à informations incomplètes modélise des situations dans lesquelles certains joueurs ont une certaine information privée avant le début d'un jeu. Cette information initiale privée, appelée "type d'un joueur", décrit complètement n'importe quelle information, non commune, que possède un joueur.
Un joueur peut avoir plusieurs types, un pour chacune de ses actions possibles. Il est également supposé que chaque joueur connaît son propre type avec une certitude absolue.
Nous passerons maintenant à la déscription des différents aspects de ce modèle.

3.2.1 Modélisation

A travers la définition des jeux de signalisation donnée dans le chapitre précédent, la détection d'intrusions dans les MANETs peut être modélisée en tant que telle pour un certain nombre de raisons, qui sont [18] :

1. Tout d'abord, dans un environnement de MANET, il est très difficile de distinguer un ami d'un ennemi en l'absence de mécanismes de sécurité. Par conséquent, le type d'un nœud particulier n'est pas facilement vérifiable par d'autres nœuds dans le système.
2. Deuxièmement, un IDS répond à l'intrusion après qu'elle ait eu lieu. Par conséquent, la modélisation de détection d'intrusion dans un cadre théorique de jeu basé sur les jeux dynamiques non-coopératifs est la bonne direction à prendre.

La modélisation faite dans [18] est décrite ci-dessous.

- **Joueurs**
  Le jeu se déroule entre deux joueurs : l'attaquant noté 1 et l'IDS noté 2. Comme nous le savons, il n'existe pas qu'un seul attaquant susceptible d'attaquer le réseau, cependant le modèle suppose qu'une seule attaque à la fois et qu'il n'existe pas de collusion entre les attaquants malveillants, ce qui rend le jeu à deux joueurs parfaitement adéquat dans ce genre de cas.
- **Stratégies**

  – Joueur 1 (attaquant) : l'information privée du joueur 1 est sa nature, ce qui veut dire qu'il peut avoir deux types : nœud régulier ou bien nœud malveillant. Ainsi, l'espace des types du joueur 1 est :

  $$\Theta = \{\text{régulier, malveillant}\}.$$

  Le joueur 1 possède des stratégies pour chacun de ces types.

  1. Régulier : dans ce cas, l'ensemble de stratégies du joueur 1 est composé d'un seul élément qui est de ne pas attaquer, c'est à dire adopter un comportement normal,
  2. Malveillant : ici le joueur 1 possède deux stratégies :

     $$\{\text{attaquer, ne pas attaquer}\}.$$

     Ces deux stratégies sont choisies avec des probabilités s et $1 - s$, respectivement.

– Joueur 2 (IDS) : l'ensemble de stratégies de l'IDS est constitué de deux éléments :

{détecter l'attaque, manquer l'attaque}.

L'IDS détecte une attaque avec une probabilité t et manque cette attaque avec $1-t$. Ces probabilités dépendent de ses croyances à priori concernant le comportement du joueur 1.

- **Gains**

Les gains des deux joueurs sont les suivants :

– L'attaquant possède un gain de δ_{intrus} lors d'une attaque non-détectée et un coût de δ_{pris} dans le cas où l'attaque a été détectée.
Ainsi, son gain attendu dans tous les cas est :

$$s[(1-t)\delta_{intrus} - t\delta_{pris}]. \tag{3.1}$$

L'attaquant essayera constamment de maximiser (3.1).
Dans le cas d'une fausse alarme, (ne pas attaquer, détecte), l'attaquant aura un gain de 0.

– L'IDS a un gain de $\gamma_{défendre}$ lorsqu'il détecte une attaque et des coûts de $\gamma_{manquer}$ et $\gamma_{falarme}$ lorsqu'il en manque une et lorsqu'il s'agit d'une fausse alerte, respectivement.
Le gain attendu pour l'IDS est alors :

$$st\gamma_{defendre} - s(1-t)\gamma_{manquer} + (1-s)t\gamma_{falarme},$$

$$\Rightarrow st\gamma_{defendre} - s\gamma_{manquer} + st\gamma_{manquer} - t\gamma_{falarme} + st\gamma_{falarme},$$

$$st(\gamma_{defendre} + \gamma_{falarme} + \gamma_{manquer}) - s\gamma_{manquer} - t\gamma_{falarme}. \tag{3.2}$$

L'IDS essayera toujours de maximiser (3.2).

Le critère de performance d'un IDS est le taux de fausses alarmes dans le système, plus ce dernier augmente plus la performance de l'IDS diminue.

3.2.2 Procédure de résolution du jeu

L'Equilibre de Nash pour chaque jeu de signalisation est décrit par les conditions suivantes :

– Pour un nœud attaquant/régulier il faudrait :

1. Donner les stratégies de l'IDS ;
2. Pour chaque type θ d'un nœud, évaluer l'utilité d'un message envoyé s_i^1 de la manière suivante

$$\sum_{j=1}^{n_2} \sigma_2(s_j^2|a_i^1)u_1(s_i^1, s_j^2, \theta).$$

– Pour l'IDS, Il procédera en deux étapes :

1. **Etape 1**
   Pour chaque action a_i^1 de l'expéditeur dont le type est θ, l'IDS utilise la règle de Bayes pour calculer la probabilité à posteriori $\mu(\theta|s_i^1)$.
2. **Etape 2**
   En utilisant les croyances calculées par la règle de Bayes, pour chaque action s_i^1 de l'expéditeur dont le type est θ, l'IDS choisit une action s_i^2 qui doit être la meilleure réponse pour cette action là.
   Par conséquent, on peut dire d'une action, s_i^2, de l'IDS qu'elle sera une meilleure réponse à une action de l'expéditeur si et seulement si elle maximise son utilité espérée sur l'ensemble de toutes les stratégies pures possibles.

Le choix d'une stratégie doit être basée sur les croyances à priori du récepteur et elle doit maximiser le gain et cela en minimisant le coût dû aux fausses alertes et aux attaques manquées.
Dans cette section nous avons présenté le modèle d'interaction entre un IDS et un attaquant sous forme d'un jeu dynamique non coopératif à information incomplète. Le modèle pourrait être étendu en prenant en compte les groupes de collusions des attaquants.

3.3 Jeux Bayésiens dans la sécurité des réseaux ad hoc

La modélisation de l'interaction entre un attaquant et un défenseur dans un réseau peut aussi se faire en se basant sur d'autres types de jeux. Nous trouvons dans les travaux [22, 7], l'application des jeux Bayésien. Dans cette section, nous allons présenter le modèle proposé dans [22].

3.3.1 Modélisation

Les auteurs dans [22] ont modélisé l'interaction entre l'IDS et l'attaquant par un jeu Bayésien à deux joueurs, dont les composants sont :

- **Les joueurs :** le jeu se déroule entre deux joueurs, l'attaquant noté par i et l'IDS noté par j ;
- **Les types :**
  Le joueur i possède deux types :
  1. Type régulier noté par $\theta_i = 0$,
  2. Type malveillant noté par $\theta_i = 1$;

  Ces types représentent l'information privée de l'attaquant que l'IDS ignore.
  Le joueur j possède un seul type qui est régulier, $\theta_j = 0$, et il est connu par les deux joueurs, i et j ;
- **Stratégies :** l'IDS n'a pas besoin d'être activé pendant tout le temps au cours duquel le réseau ad hoc est en place. Ainsi, l'ensemble de ses stratégies pures

est :
S_j = {surveiller pour un temps t, ne pas surveiller}, t∈[0, 1]. La première stratégie représente le cas où l'IDS est activé pour un certain temps noté t. Donc, l'IDS surveille périodiquement le trafic et le reste du temps il reste inactif. De même pour l'attaquant, il n'a pas besoin d'attaquer tout le temps. L'ensemble de ses stratégies pures est :

1. Pour le type malveillant c'est :
   S_i = {attaquer pour un temps s, ne pas attaquer},
2. Le type régulier ne possède qu'une seule stratégie, qui est ne pas attaquer ;

Pour avoir les profits des deux joueurs, i et j, pour chaque stratégie possible, on a besoin de définir les gains et les pertes qui leur sont associés :

- a : représente le taux de détection d'une intrusion par le joueur j, a∈[0, 1] ;
- b : représente le taux d'une fausse alarme pour le joueur j, b∈[0, 1] ;
- m : représente le gain du joueur j pour une détection réussie ;

l : représente la perte du joueur j pour ne pas avoir détecté l'attaque durant toute la durée ;

- n : représente la perte causée par une fausse alarme pour le joueur j ;
- c_a : représente le coût d'une attaque du joueur i durant toute la période ;
- c_d : représente le coût de surveillance pour le joueur j durant toute la période ;

Avec : m ≥ l et l ≥ c_a, c_d, ces suppositions paraissent raisonnables car sinon les deux joueurs i et j ne seront pas incités à attaquer et à surveiller, respectivement.

Comme nous l'avons déjà défini, le joueur j surveille le réseau pour un certain temps noté t et le joueur i attaque pour un temps s. Ainsi, la probabilité que l'IDS soit en état de surveillance au moment où l'attaque se produit est donnée par **st**, le gain associé au joueur j dans ce cas est **mst**. En tenant compte du taux de détection, a, le gain sera :

$$amst - (1 - a)mst.$$

La probabilité que l'IDS soit inactif au moment où l'attaque se produit est donnée par **(1 - t)s**, en raison de laquelle le joueur j perd :

$$(1 - t)sl.$$

Nous avons tc_d c'est le coût de surveillance, ainsi, le gain du joueur j d'avoir réussi à détecter une attaque est :

$$amst - (1 - a)mst - (1 - t)sl - tc_d = (2a - 1)mst - (1 - t)sl - tc_d.$$

Dans le cas ou il est inactif il sera égal à :

$$-sl.$$

Une fausse alerte engendrera une perte pour le joueur j égale à :

$$btn,$$

donc le gain dans ce cas sera :

$$-btn - tc_d.$$

Les gains du joueur i s'il choisit la stratégie ne pas attaquer, seront égaux à 0 quelle que soit la stratégie choisie par le joueur j et dans le cas contraire, ils seront égaux aux pertes du joueurs j en soustrayant le coût engendré par l'attaque, sc_a.

Ainsi, les profits des deux joueurs i et j, l'attaquant et l'IDS, pour les deux différents types du joueur i, $\theta_i = 1$ et $\theta_i = 0$, sont résumés dans les tableaux suivants :

$\theta_i = 1$:

i\ j	$S_j(1)$	$S_j(2)$
$S_i(1)$	((1 - 2a)mst + (1 - t)sl - sc_a , (2a - 1)mst - (1 - t)sl - tc_d)	(sl - sc_a, -sl)
$S_i(2)$	(0, -btn - tc_d)	(0, 0)

$\theta_i = 0$:

i\ j	$S_j(1)$	$S_j(2)$
$S_i(2)$	(0, -btn - tc_d)	(0, 0)

où :

- $S_i(1)$: attaquer pour un temps s ;
- $S_i(2)$: ne pas attaquer ;
- $S_j(1)$: surveiller pour un temps t ;
- $S_j(2)$: ne pas surveiller.

3.3.1.1 Equilibre de Nash Bayésien

L'objectif des deux joueurs i et j est de maximiser leurs propres profits. Comme on l'a déjà cité l'attaquant peut être de type malveillant ou bien régulier. Ainsi, soit μ_0 la probabilité qu'il soit malveillant. Le jeu peut être représenté sous forme extensive comme le montre la figure suivante :

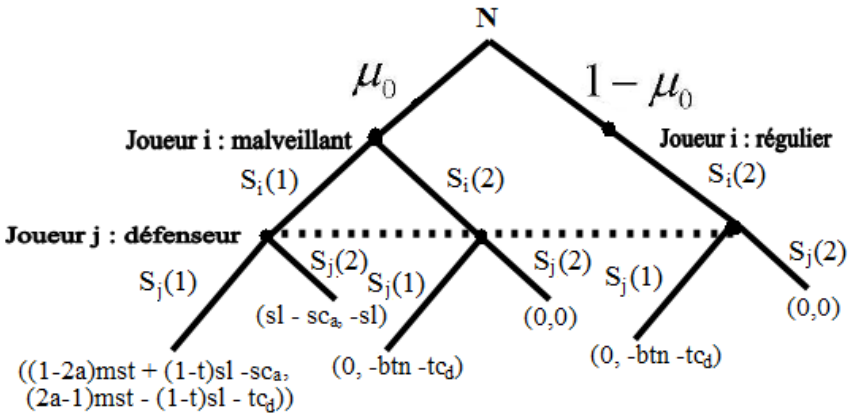


FIGURE 3.1 – Jeu sous forme extensive

Les gains espérés du joueur j dans ce cas, pour ses deux stratégies $S_j(1)$ et $S_j(2)$, seront :

$$E_j(S_j(1)) = \mu_0((2a-1)mst - (1-t)sl - tc_d) - (1-\mu_0)(btn + tc_d),$$

$$E_j(S_j(2)) = -\mu_0 sl.$$

Si $\mu_0 > \frac{bn+cd}{(2a-1)sm+sl+bn}$, $E_j(S_j(1)) > E_j(S_j(2))$, alors la meilleure stratégie pour le joueur j est de surveiller pour un temps t. Cependant, si le joueur j joue cette stratégie, la meilleure stratégie du joueur i n'est pas d'attaquer pour un temps s mais plutôt de ne pas attaquer. D'où, le couple de stratégies :
((Attaquer pour un temps s, si i est malveillant, ne pas attaquer, si i est régulier), (surveiller pour un temps t, μ_0)) ne constitue pas un équilibre de Nash Bayésien.

Si $\mu_0 < \frac{bn+cd}{(2a-1)sm+sl+bn}$, alors le couple de stratégies ((Attaquer pour un temps s, si i est malveillant, ne pas attaquer, si i est régulier), (ne pas surveiller pour un temps t, μ_0)) constitue un équilibre de Nash Bayésien, car on a :

- -sl > (2a - 1)mst - (1 - t)sl - tc_d, ceci signifie que : si le joueur i maintient sa stratégie $S_i(1)$, le second n'a pas intérêt de dévier ;
- sl - sc_a = s(l - c_a) > 0, puisque l est supposé supérieure à c_a, ceci signifie que : si le joueur j maintient sa stratégie $S_j(2)$, le premier n'a pas intérêt de dévier.

Et :

- 0 > -btn - tc_d, c'est pour le cas, où le joueur i est régulier.

Nous avons que si $\mu_0 > \frac{bn+cd}{(2a-1)sm+sl+bn}$, l'équilibre de Nash Bayésien n'existe pas en stratégies pures, mais il existe en stratégies mixtes. Soient alors p et q les probabilités avec lesquelles les joueurs i et j jouent leurs premières stratégies respectivement. Ainsi, les gains du joueur j pour ses deux stratégies, $S_j(1)$ et $S_j(2)$ seront :

$$E_j(S_j(1)) = p\mu_0((2a-1)mst - (1-t)sl - tc_d) - (1-p)(1-\mu_0)(btn + tc_d),$$

$$E_j(S_j(2)) = -p\mu_0 sl.$$

Pour $E_j(S_j(1)) = E_j(S_j(2))$, nous obtenons que la stratégie d'équilibre du joueur i de type malveillant est de jouer $S_i(1)$ avec la probabilité :

$$p^* = \frac{bn + c_d}{\mu_0((2a-1)sm + sl + bn)}.$$

Et pour le second joueur, ils seront :

$$E_i(S_i(1)) = q((1-2a)mst + (1-t)sl - sc_a) + (1-q)(sl - sc_a),$$

$$E_j(S_j(2)) = 0.$$

Pour $E_i(S_i(1)) = E_i(S_i(2))$, nous obtenons que la stratégie d'équilibre du joueur j est de jouer $S_j(1)$ avec la probabilité :

$$q^* = \frac{l - c_a}{(2a-1)tm + tl}.$$

Ainsi, le couple de stratégies ((Attaquer pour un temps s avec une probabilité p^*, si i est malveillant, ne pas attaquer, si i est régulier), (surveiller pour un temps t avec une probabilité q^*, μ_0)) constitue un équilibre de Nash Bayésien en stratégies mixtes.

Comme nous l'avons déjà mentionné, la modélisation de l'interaction entre un IDS et un attaquant sous forme d'un jeu Bayésien a fait aussi l'objet de l'article [7], où les auteurs n'ont pas pris en compte le temps de surveillance de l'IDS et le temps d'une attaque. Par conséquent, le modèle présenté ci-dessus est une extension de celui donné en [7] en ajoutant des hypothèses sur les stratégies de chacun des joueurs, l'IDS et l'attaquant.

Un inconvénient possible pour ce modèle est dans la pratique, il peut être difficile à déterminer une probabilité antérieure raisonnable μ_0. Dans des applications pratiques, le défenseur peut assigner μ_0 basé sur sa connaissance de l'environnement du réseau.

Le modèle présenté dans cette section modélise le cas où il n'existe qu'une seule attaque et donc l'IDS réagit qu'une seule fois, nous pourrions considérer le cas où les attaques se répètent ce qui amène l'IDS à réagir à chacune d'elles, dans ce cas l'application d'un jeu de signalisation serait adéquate.

Dans [24], nous trouvons une autre modélisation de l'interaction entre un attaquant et un défenseur dans un réseau. Comme les attaques et défenses changent l'état du celui-ci, alors la modélisation idéale pour analyser les propriétés de ces interactions était sous forme d'un jeu stochastique à somme non nulle. Cette classe de jeu a été introduite et étudiée pour la première fois par Loyd Shapley (1953).

3.4 Conclusion

Ce chapitre a été consacré au regard de la littérature existante sur l'application de la théorie des jeux pour les problèmes de sécurité informatique. Par conséquent,

nous avons pu constater que la modélisation de l'interaction entre un attaquant et un IDS dans un réseau pouvait se faire sous différents types de jeu. Les modélisations présentées nous ont permises d'avoir un regard nouveau sur les problèmes de sécurité, la manière de les traiter et de les résoudre.

Le prochain chapitre sera dédié à la présentation du modèle que nous avons élaboré où l'on illustre le problème de sécurité des réseaux ad hoc par la théorie des jeux évolutionnaires.

4

Modèle du jeu pour la sécurité des réseaux ad hoc

4.1 Introduction

Le problème de sécurité dans les réseaux ad hoc a longtemps été l'un des soucis majeurs lors de la conception d'un réseau. Plusieurs approches ont été mises en avant dans le but de contrecarrer les menaces qui pèsent sur ce type de réseaux. Dans le chapitre précédent, nous avons présenté l'une de ces approches qui est la théorie des jeux et qui a pris place dans différents travaux de sécurité avec laquelle une modélisation complète est effectuée.

L'objectif de ce chapitre est de présenter notre modèle de sécurité pour les réseaux ad hoc, en exposant les différentes étapes qui le constituent. Ensuite, nous passerons à la description du simulateur que nous avons conçu à l'aide du logiciel Matlab dans le but d'évaluer l'efficacité du modèle et ceci en effectuant quelques expériences.

4.2 Motivation

Les réseaux ad hoc sont devenus une technologie passionnante et importante ces dernières années en raison de la prolifération rapide des appareils sans fil. Avec leurs caractéristiques bien particulières qui sont l'absence d'une gestion centrale, la nature mobile de leur environnement et de leurs topologies dynamiques, ces réseaux sont plus vulnérables et plus difficiles à protéger que les réseaux filaires. En premier lieu, l'utilisation de l'interface radio pour les communications rend ces réseaux susceptibles aux différentes attaques. De plus, la mise en œuvre de certains mécanismes de sécurité développés pour les réseaux filaires est délicate, voire impossible dans les réseaux ad hoc. En raison de leur caractère spontané, ces derniers ne peuvent béné-

ficier des mécanismes de sécurité s'appuyant sur l'infrastructure, comme un pare-feu ou un serveur d'authentification [2]. Le principal problème de ce type de réseaux ne se situe pas seulement au niveau du support physique, mais principalement dans le fait que tous les nœuds sont équivalents et potentiellement nécessaires au fonctionnement du réseau. Ainsi, les possibilités de s'insérer dans le réseau sont plus grandes et la détection d'une intrusion ou d'un déni de service est plus délicate.

Afin de maintenir la sécurité d'un réseau ad hoc, tous les nœuds le composant doivent collaborer en participant au mécanisme de défense du réseau et cela en activant leur système de détection d'intrusion (IDS) installé au préalable. Cependant, un nœud sera protégé s'il participe avec un nombre minimum de nœuds (seuil) au mécanisme de défense. Si ce seuil n'est pas atteint, le nœud activant son IDS perdra plus qu'un autre nœud dont l'IDS est inactif car un nœud qui participe à la sécurité dépense des ressources et finit par ne pas être protégé. Toutefois, si le seuil est atteint, il est préférable pour tous les nœuds de participer à ce mécanisme de défense pour se protéger, ainsi, le réseau sera sécurisé. En conséquence, si un nœud a une forte confiance que les autres nœuds vont choisir de participer, il participera également, sinon il ne doit pas participer.

Ce scénario décrit le lien étroit entre la confiance, la coopération et la sécurité. Nous trouvons, dans [12], la modélisation de la sécurité dans les réseaux ad hoc comme une interaction stratégique. Toute décision d'un nœud, participer ou non dans le mécanisme de sécurité, affecte la décision des autres nœuds et le résultat final qui est la sécurité du réseau entier. La théorie des jeux est la branche des mathématiques appliquées qui formalise l'interaction stratégique entre des agents autonomes rationnels. Si les agents ne sont pas supposés être rationnels, la théorie des jeux évolutionnaires est plus appropriée. Dans [12], la théorie des jeux et la théorie des jeux évolutionnaires ont été utilisées pour modéliser la confiance entre les nœuds et voir la dynamique évolutionnaire du comportement de confiance des nœuds, respectivement.

Les nœuds d'un réseau ad hoc sont caractérisés par des ressources limitées en énergie et comme les solutions de sécurité sont gourmandes en terme de ressources (mémoire et surtout énergie) [19], ça ne sera pas toujours efficace de faire de tous les nœuds des nœuds de contrôle [1]. Dans le but de surmonter ce problème, nous proposons un modèle où l'on suppose que les IDS ne sont activés que sur un nombre restreint de nœuds du réseau. Néanmoins, ces nœuds peuvent ne pas accomplir leur tâche de contrôle en désactivant leur IDS. Afin de voir l'évolution de l'état du réseau qui émerge des attitudes des nœuds sélectionnés pour assurer le contrôle du trafic dans le réseau, nous allons élaborer un modèle en se basant sur le modèle proposé dans [12] et en ajoutant un paramètre concernant les dommages et les pertes causés dans le cas où le réseau n'est pas sécurisé.

1. Un nœud de contrôle est un nœud dont l'IDS est activé.

4.3 Présentation du modèle

Nous considérons un réseau ad hoc composé d'une population de nœuds, notre modèle sera alors basé sur une démarche à deux étapes : clusteristaion et modélisation sous forme de jeu.

4.3.1 Clusterisation

En premier lieu, une technique de clusterisation sera appliquée au réseau et qui consistera à le partitionner virtuellement en groupant l'ensemble des nœuds en clusters. Ainsi, un cluster head est élu et il sera utilisé pour prendre le rôle de nœud de contrôle pour l'ensemble des nœuds de son groupe, c'est à dire prendre en charge le contrôle de tout le trafic destiné aux membres de son cluster. Comme nous l'avons vu dans le premier chapitre, il existe plusieurs algorithmes de clusterisation et chacun d'eux se base sur une métrique particulière ou une combinaison de métriques. Dans cette section, nous allons proposer un algorithme de clusterisation tout en prenant en compte deux métriques dans l'élection du cluster head : le niveau d'énergie et le degré de chaque nœud et ceci pour les raisons suivantes :

- Le cluster head doit avoir un degré maximum pour générer un nombre réduit de clusters, ce qui implique un nombre réduit d'IDS actifs et donc une conservation d'énergie plus longue ;
- Le niveau d'énergie requis pour un cluster head doit être conséquent afin de garder son IDS actif le plus longtemps possible durant la période pendant laquelle il est cluster head, ce qui augmente la sécurité du réseau.

4.3.1.1 Algorithme de Clusterisation

L'algorithme de clusterisation proposé est décrit comme suit :

Algorithm 5 Algorithme de clusterisation

(0). Initialisation : l'étape d'initialisation consiste à :

1. Affecter pour chaque nœud une position de départ et lui fixer une charge d'énergie initiale,
2. Fixer le niveau d'énergie requis pour qu'un nœud soit prioritaire dans le processus d'élection des cluster heads ;

(1). Calcul des degrés : dans cette étape, nous calculons le degré de chaque nœud qui correspond au nombre de ses voisins à un saut ;
(2). Test d'énergie : cette étape consiste à sélectionner un ensemble S, qui contiendra les nœuds vérifiant le seuil fixé dans l'étape d'initialisation ;
(3). Désignation des cluster heads : en arrivant à cette étape, l'ensemble S peut être vide ou contenant au moins un élément :

Si $S \neq \emptyset$:

1. le principe de l'algorithme HCC sera appliqué, où l'on sélectionne le nœud dont le degré est le plus élevé dans l'ensemble S, que nous noterons CH,
2. S=S\{CH} ;

Sinon, si $S = \emptyset$ dans ce cas, nous appliquerons l'algorithme HCC sur l'ensemble des nœuds restants ;

(4). Rattachement aux cluster heads : une fois que le cluster head est élu, tous ses voisins à un saut le rejoignent et ainsi ils formeront un cluster.
Les étapes 3 et 4 seront répétées jusqu'à ce que tous les nœuds soient affiliés aux clusters que ce soit en tant que cluster head ou bien membre.

Comme les nœuds d'un réseau ad hoc sont dynamiques et à énergie limitée, il serait indispensable de mettre à jour la procédure de clusterisation ce qui fait que chaque nœud est potentiellement capable d'être élu cluster head. Il est donc nécessaire d'installer des IDS sur tous les nœuds du réseau et de les garder en veille jusqu'au moment où ils seront élus cluster heads.

Nous passerons à présent à la description du modèle du jeu.

4.3.2 Modèle du jeu

Une fois que les cluster heads sont élus, ces derniers auront le choix entre activer leur IDS afin d'assurer la sécurité de l'ensemble des nœuds de leur cluster ou bien de le désactiver dans le but de conserver l'énergie consommée par celui ci. Cette situation peut être modélisée sous forme de jeu entre les cluster heads. Comme les nœuds d'un réseau disposent d'une rationalité limitée et ils ont les mêmes aptitudes, il serait plus adéquat d'appliquer les jeux évolutionnaires. Ainsi, le jeu se déroulant entre les cluster heads du réseau sera décrit comme suit :

- **Joueurs :** l'ensemble des joueurs est l'ensemble des cluster heads du réseau ad hoc et les compétitions se feront par paire de cluster heads tirés de manière aléatoire ;

- **Stratégies :** les cluster heads ont le même ensemble de stratégies qui est {Protéger (P), Ne pas Protéger (NP)} ;
- **Gains :** un nœud qui contribue à la sécurité, donc opter pour la stratégie P :
  - Supporte un coût égal à c qui correspond à la dépense énergétique lorsqu'il assure la sécurité de son cluster,
  - Et obtient une récompense r lorsque tout le réseau est sécurisé, c'est à dire tous les cluster heads choisissent la stratégie P,
  - Lorsqu'au moins un cluster head choisit de ne pas activer son IDS, le réseau devient non sécurisé, car il est supposé être constamment exposé aux attaques. Ce fait engendrera des pertes de données et des dommages que nous noterons l.

Ainsi, le jeu entre deux cluster heads peut être représenté sous forme stratégique, avec les gains associés à chaque couple de stratégies, comme suit :

	P	NP
P	(r-c, r-c)	(-c-l, -l)
NP	(-l, -c-l)	(-l, -l)

TABLE 4.1 – Forme stratégique du jeu Protéger-Ne pas Protéger

où : r, c et $l > 0$.
Nous supposons que $r > c$ et $l > c$, car sinon les cluster heads ne seront pas incités à protéger le réseau.
En considérant la configuration ci-dessus, nous passerons à l'étude du jeu et ceci en cherchant l'ESS en stratégies pures.

4.3.2.1 Recherche des ESS

Comme le jeu étudié est un jeu symétrique, nous allons considérer la même matrice des gains :

$$A = \begin{matrix} \\ P \\ NP \end{matrix} \begin{matrix} \begin{matrix} P & \quad NP \end{matrix} \\ \begin{pmatrix} r-c & -c-l \\ -l & -l \end{pmatrix} \end{matrix}$$

Nous remarquons que :

- u(P, P) > u(NP, P), car (r-c) > -l ⇒ (P, P) est un équilibre de Nash strict, d'après la proposition 2.6.2, P constitue une ESS ;
- u(NP, NP) > u(P, NP), car -l >(-c-l) ⇒ (NP, NP) est un équilibre de Nash strict, d'après la proposition 2.6.2, NP constitue une ESS.

Par conséquent, le jeu modélisant la situation décrite précédemment admet deux ESS.
Dans ce modèle, nous avons considéré une population constituée de deux groupes caractérisés par deux stratégies pures P et NP. Dans le but d'analyser la façon dont ces groupes évoluent au cours du temps, nous allons introduire le concept du réplicateur dynamique donné par la formule (2.12).

4.3.2.2 Réplicateur dynamique

Dans cette section nous allons prendre en compte l'histoire des stratégies car même si, au moment de jouer, les cluster heads ont déterminé la stratégie à adopter, ils possèdent assez d'information pour décider de la changer, si elle s'avère inefficiente. La théorie des jeux évolutionnaires interprète l'inefficience comme le fait qu'une stratégie offre un gain inférieur au gain moyen qu'offre l'ensemble des stratégies.
Soit p = (p_1, p_2), tel que p_1 est la proportion de joueurs choisissant la stratégie P (protéger) et donc p_2 = 1-p_1 représentera la proportion de ceux qui ne protègent pas, optant pour NP.
En appliquant la formule du réplicateur dynamique définie comme suit :

$$\dot{p_i} = p_i[(Ap)_i \ - \ p^T Ap], \quad i = 1, 2.$$

Avec :

$$\begin{aligned} Ap = \begin{pmatrix} r-c & -c-l \\ -l & -l \end{pmatrix} \begin{pmatrix} p_1 \\ 1-p_1 \end{pmatrix} &= \begin{pmatrix} p_1(r-c)+(1-p_1)(-c-l) \\ -l \end{pmatrix} \\ &= \begin{pmatrix} p_1(r+l)-c-l \\ -l \end{pmatrix} \end{aligned}$$

Et :

$$\begin{aligned} p^T Ap = (p_1, (1-p_1)) \begin{pmatrix} p_1(r+l)-c-l \\ -l \end{pmatrix} &= p_1^2(r+l) - p_1 c - p_1 l - l(1-p_1) \\ &= p_1^2(r+l) - p_1 c - l. \end{aligned}$$

Nous aurons :

$$\begin{aligned} \dot{p_1} &= p_1[p_1(r+l) - c - l - p_1^2(r+l) + p_1 c + l] \\ &= p_1(1-p_1)[p_1(r+l) - c]. \end{aligned}$$

Donc :

$$\dot{p_1} = p_1(1-p_1)[p_1(l+r) - c]. \tag{4.1}$$

Passons à présent au calcul de $\dot{p_2}$

$$\dot{p_2} \ = \ p_2[-l - p_1^2(r+l) + p_1 c + l].$$

En remplaçant p_1 par 1-p_2, nous aurons :

$$\begin{aligned} \dot{p_2} &= p_2[-(1-p_2)^2(r+l) + (1-p_2)c] \\ &= p_2(1-p_2)[-(1-p_2)(r+l) + c]. \end{aligned}$$

Et donc :

$$\dot{p_2} = p_2(1-p_2)[p_2(r+l) - r - l + c]. \tag{4.2}$$

Comme p_2=1-p_1, l'équation (4.2) sera égale à :

$$\begin{aligned} \dot{p_2} &= p_1(1-p_1)[(1-p_1)(r+l)-r-l+c] \\ &= p_1(1-p_1)[r+l-p_1r-p_1l-r-l+c] \\ &= p_1(1-p_1)[p_1(-r-l)+c] \\ &= -\dot{p_1}. \end{aligned}$$

Calcul des points stationnaires

Pour calculer les points stationnaires, nous allons utiliser la définition 1.8.11. Comme nous avons déjà calculé $\dot{p_1}$ et $\dot{p_2}$, la fonction f s'écrit :

$$\begin{aligned} f(p) &= (\dot{p_1}, \dot{p_2}), \\ &= (\dot{p_1}, -\dot{p_1}). \end{aligned}$$

$f(p) = 0 \Rightarrow \dot{p_1}$=0, passons alors à la résolution de cette dernière équation.

$$\dot{p_1} = 0 \Rightarrow p_1(1-p_1)[p_1(r+l)-c] = 0 \Rightarrow p_1 = 0 \text{ où } p_1 = 1 \text{ où } p_1 = \frac{c}{r+l}.$$

Ainsi, les points stationnaires sont : (0, 1), (1, 0) et ($\frac{c}{r+l}$, 1-$\frac{c}{r+l}$).

Evaluation de $\dot{p_1}$

Le réplicateur dynamique dépend de valeur de p_1 :

1. $p_1 > \frac{c}{r+l}$: dans ce cas le réplicateur dynamique est positif ce qui engendre une croissance de p_1, la part de la stratégie P dans la population, donc le réplicateur dynamique converge vers la stratégie évolutionnairement stable p_1^*=1 ;
2. $p_1 < \frac{c}{r+l}$: ici le réplicateur dynamique est négatif et ceci engendre d'une décroissance de p_1, la part de la stratégie P dans la popultaion. Ainsi, contrairement au premier cas, le réplicateur dynamique converge vers la stratégie évolutionnairement stable p_1^*=0 ;
3. $p_1 = \frac{c}{r+l}$: ce point constitue un point stationnaire et donc le réplicateur dynamique est nul. Ainsi, p_1 restera inchangée.

4.4 Implémentation du réplicateur dynamique

Dans cette partie, nous allons voir l'évolution du réplicateur dynamique au cours du temps. A cet effet, il existe une fonction Matlab qui a été développée pour les équations différentielles appelée ODE23.
Cette fonction est définie comme suit :

$$[T, Y] = ode23(odefun, tspan, y_0)$$

– odefun : est le nom de la fonction créée ;

– tspan : est un vecteur qui représente l'intervalle de temps étudié ;
– y_0 : est un vecteur de conditions initiales.

Quant aux résultats, ils sont contenus dans T et Y.
Nous avons créé une interface graphique à l'aide de Matlab, où nous avons intégré le réplicateur dynamique $\dot{p}_1$.

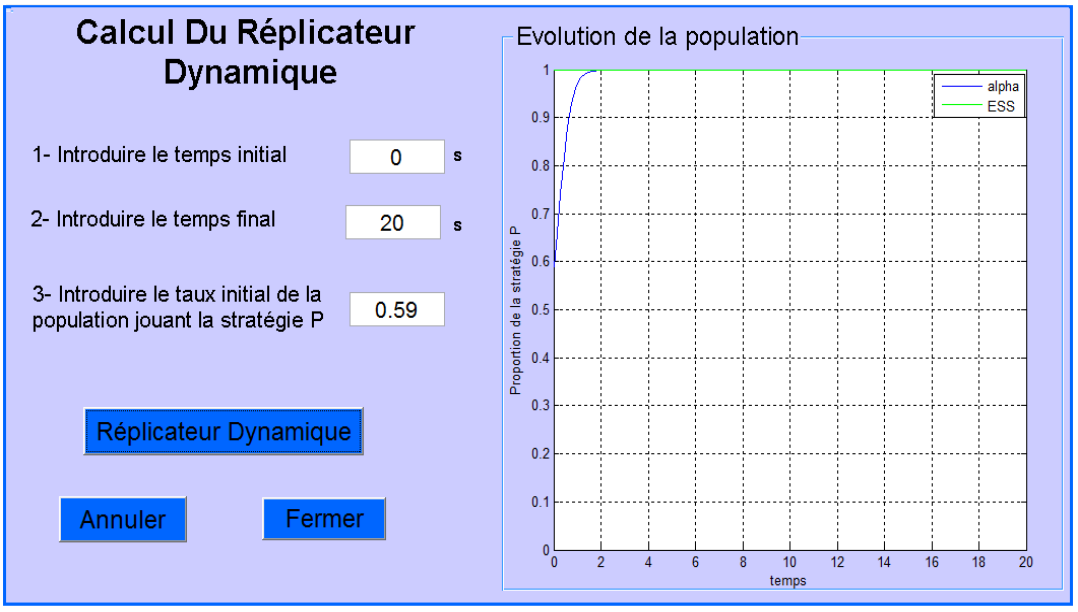


FIGURE 4.1 – Interface du réplicateur dynamique

La figure suivante montre la convergence du réplicateur dynamique de la stratégie P en faisant varier le taux initial. Les paramètres du jeu sont alors fixés à : 3, 1 et 2 pour r, c et l respectivement.

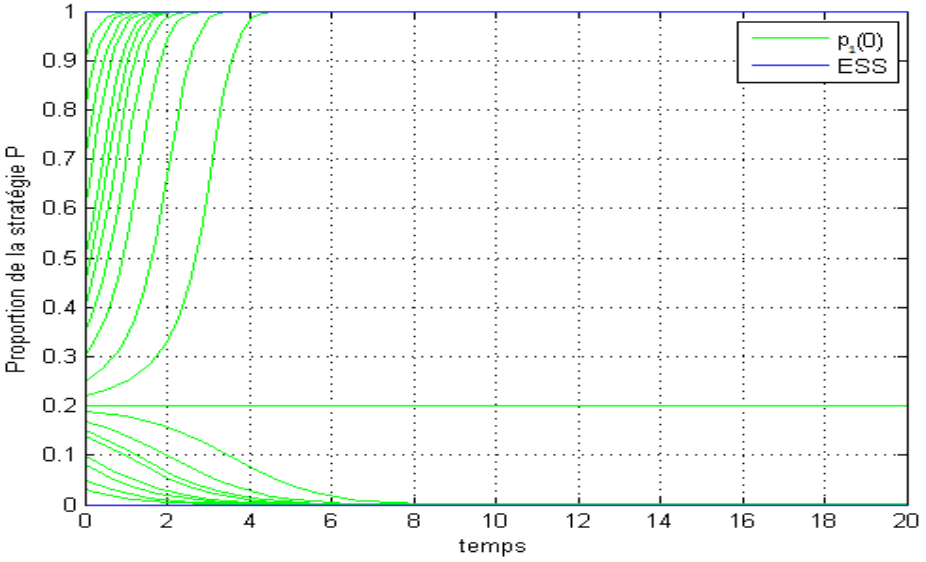


FIGURE 4.2 – Convergence du réplicateur dynamique r=3, c=1 et l=2

Nous remarquons que le replicateur dynamique converge vers les deux ESS p_1^*=1 et p_1^* =0.

Nous allons maintenant fixer les paramètres liés au jeu, puis nous ferons varier le taux de la population initiale participant à la sécurité pour étudier la convergence du réplicateur dynamique de la stratégie P vers l'ESS.

Cas 1 : variation du taux initial
Pour cette première expérience, posons r=3, c=1 et l=2. Ainsi, nous obtiendrons les résultats suivants :

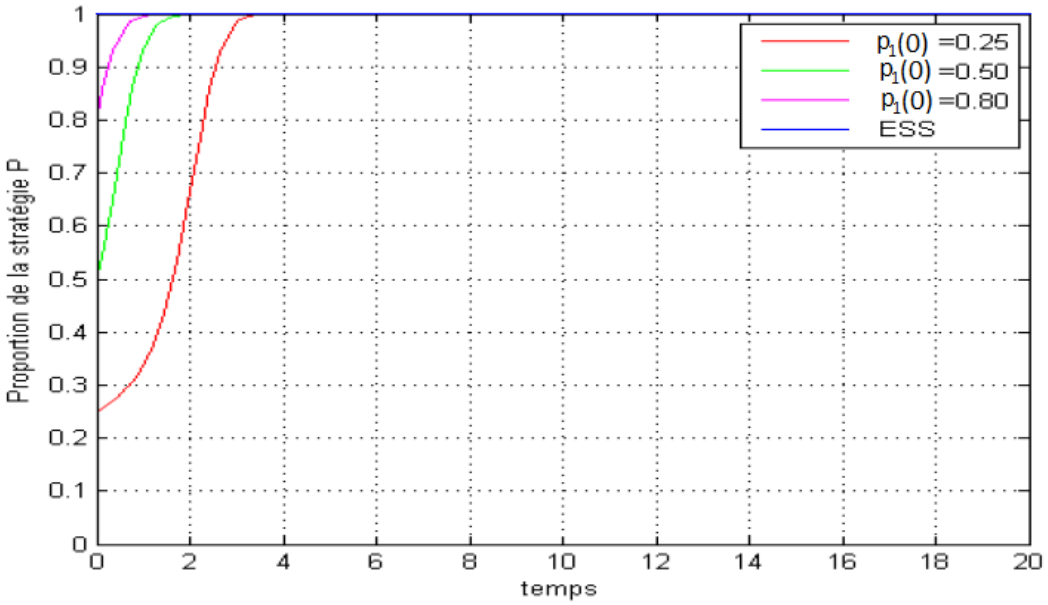


FIGURE 4.3 – Convergence du réplicateur dynamique r=3, c=1 et l=2

D'après la figure 4.3, nous voyons bien que la stratégie évolutionnairement stable p_1^*=1 est atteinte, pour $p_1(0)$=0.25, $p_1(0)$=0.50 et $p_1(0)$=0.80 à partir des instants 3, 1.8 et 1 respectivement. Nous remarquons aisément que plus la proportion initiale est proche de l'ESS, plus la rapidité de la convergence croit et donc la sécurité est assurée plus rapidement car celle ci est garantie lorsque tous les cluster heads choisissent P.

En passant à d'autres taux initiaux inférieurs à ceux de la figure 4.3 nous obtiendrons :

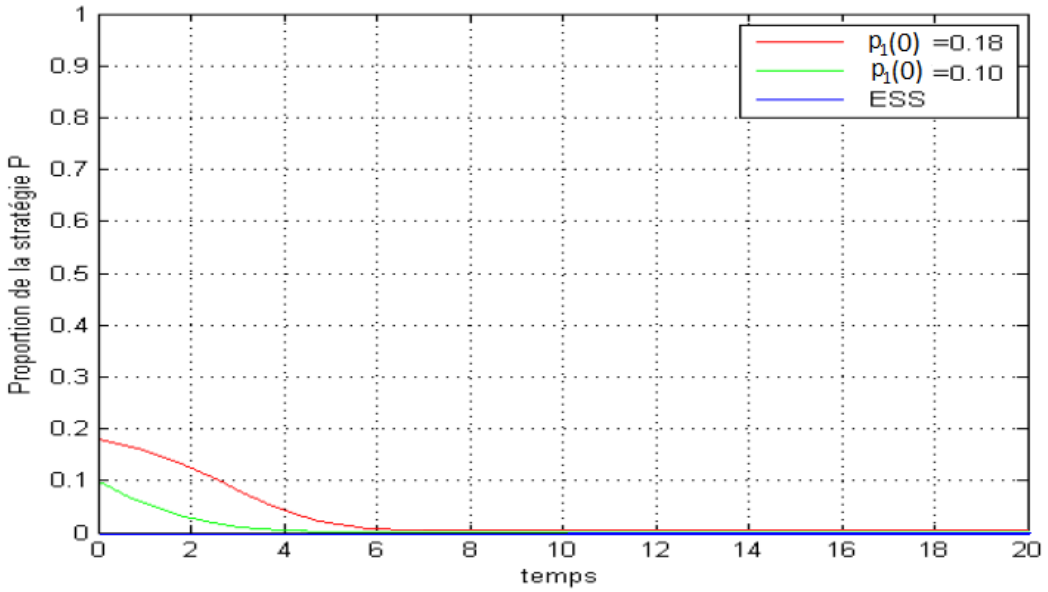


FIGURE 4.4 – Convergence du réplicateur dynamique r=3, c=1 et l=2

La figure 4.4 montre qu'avec des proportions initiales égales à $p_1(0)$=0.18 et $p_1(0)$=0.10, le réplicateur dynamique converge vers la stratégie évolutionnairement stable p_1^*=0. De même que dans la situation précédente, la vitesse de convergence du réplicateur est en relation directe avec le taux initial.
D'après ces deux expériences, nous constatons l'impact de la proportion initiale sur la convergence du réplicateur dynamique. Ce dernier peut converger alors vers les deux ESS. En effet, la sécurité du réseau est liée à l'état initial de la population.

En effectuant d'autres changements sur l'état initial, nous aurons la figure suivante :

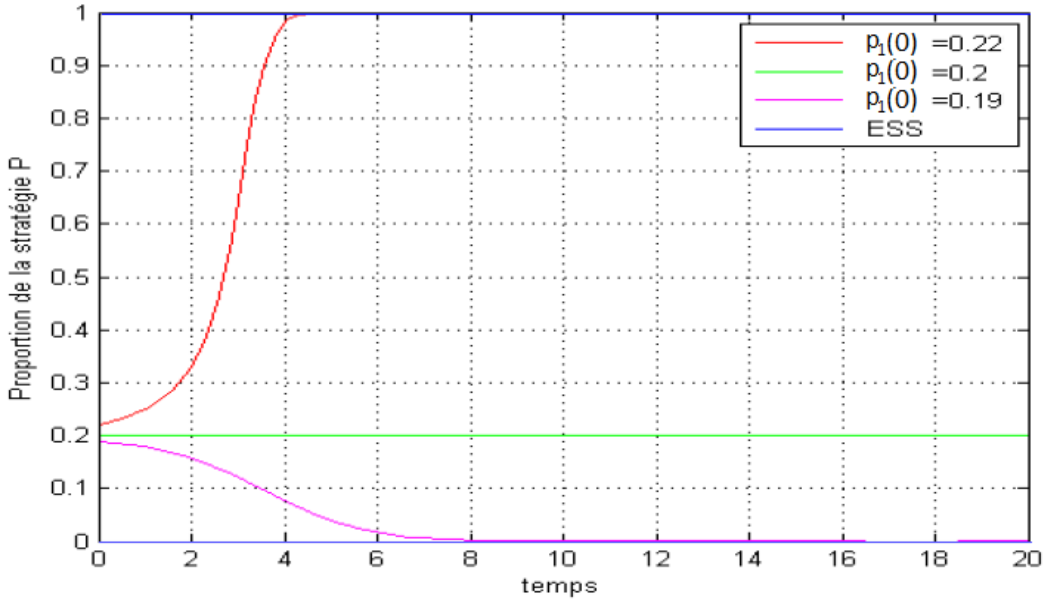


FIGURE 4.5 – Convergence du réplicateur dynamique r=3, c=1 et l=2

Nous apercevons que si l'état initial est égal à 0.2 le réplicateur dynamique demeure en cet état, ceci est dû au fait que $0.2=\frac{c}{r+l}$ est un point stationnaire comme nous l'avons mentionné dans la section précédente. Au delà de cette proportion le réplicateur dynamique converge vers la stratégie évolutionnairement stable p_1^*=1 et dans le cas inverse ça converge vers p_1^*=0.

Afin que tous les cluster heads finissent par participer au mécanisme de sécurité, il faudrait que la population initiale choisissant la stratégie P dépasse 20%. Ce qui conduit à la sécurisation du réseau.

Cas 2 : variation des paramètres du jeu

Nous passons maintenant à la variation des paramètres du jeu, tout en gardant les mêmes taux initiaux que dans le cas de la figure 4.3, $p_1(0)$=0.25, $p_1(0)$=0.5 et $p_1(0)$=0.80.

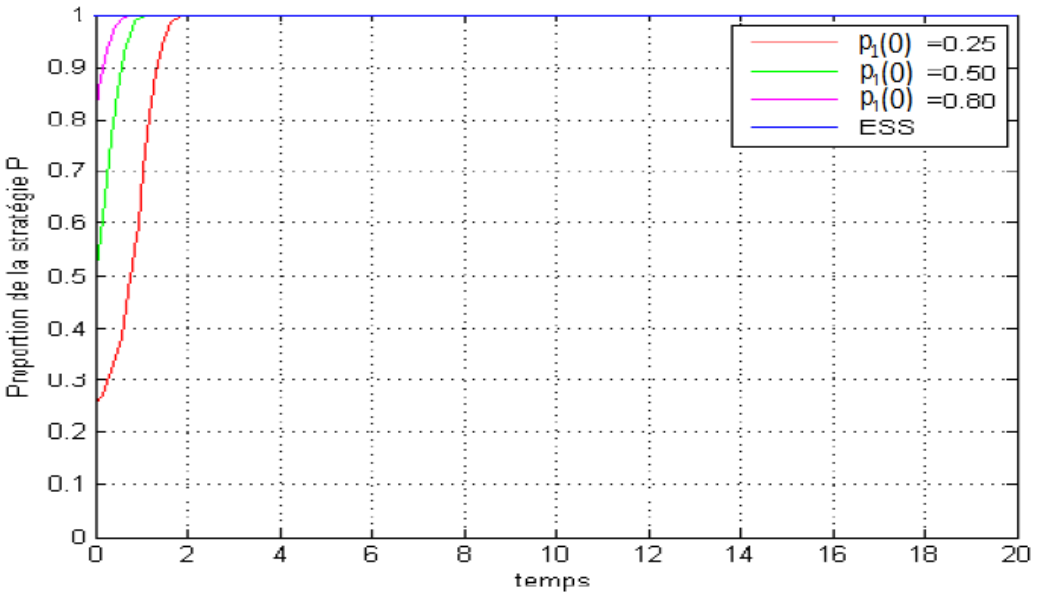


FIGURE 4.6 – Convergence du réplicateur dynamique r=5, c=1, l=2

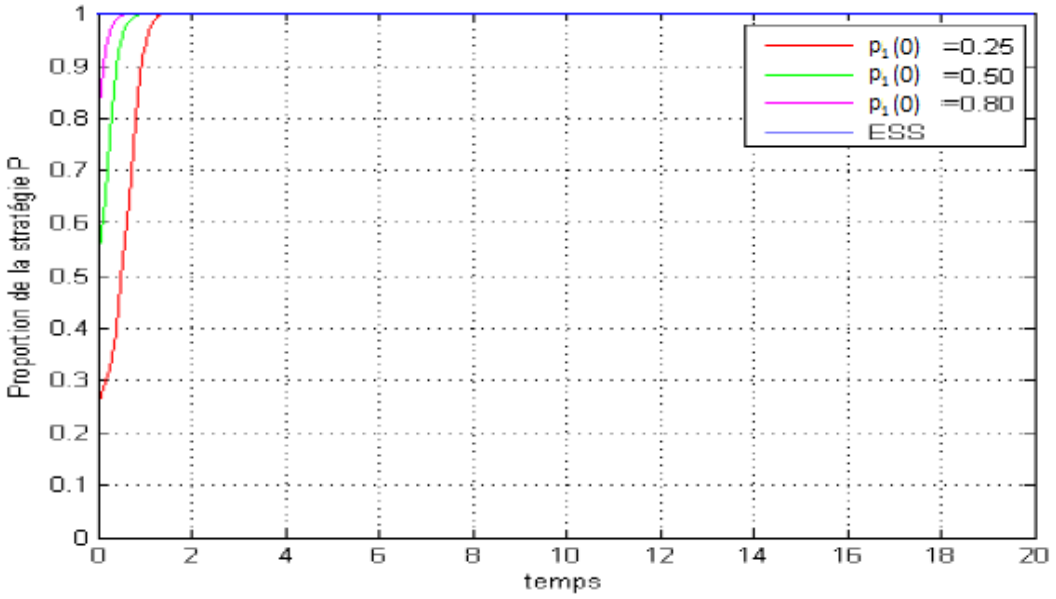


FIGURE 4.7 – Convergence du réplicateur dynamique r=5, c=1, l=4

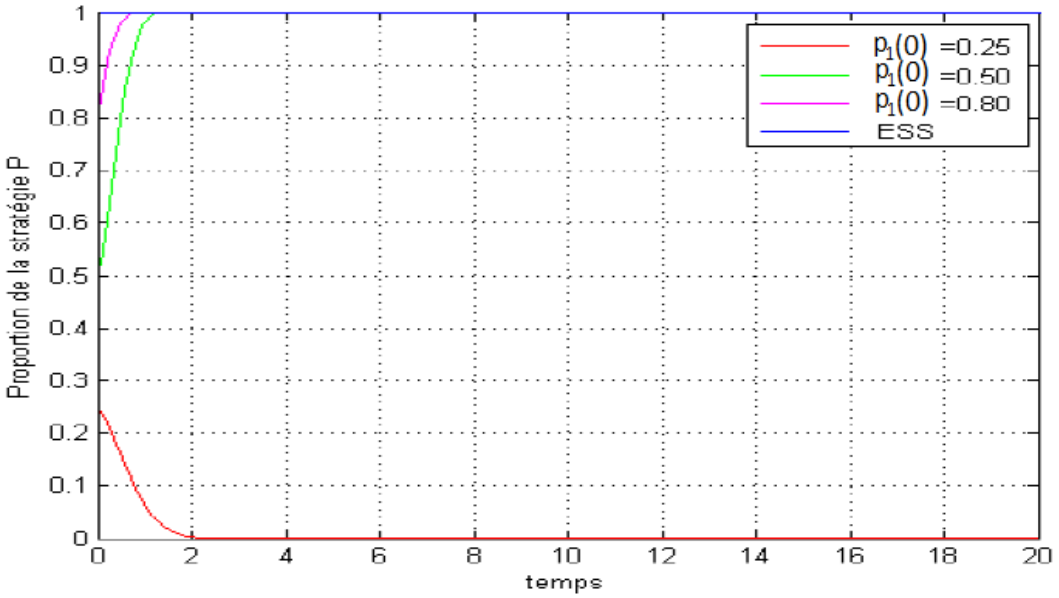


FIGURE 4.8 – Convergence du réplicateur dynamique r=5, c=3, l=4

En comparant les trois figures 4.6, 4.7 et 4.8 avec la figure 4.3, nous remarquons clairement :

- En augmentant la récompense r et en gardant les mêmes valeurs pour c et l, le réplicateur dynamique converge vers la stratégie évolutionnairement stable p_1^*=1, pour $p_1(0)$=0.25, $p_1(0)$=0.50 et $p_1(0)$=0.80 à partir des instants 2, 1 et 0.4 respectivement. Par conséquent, nous constatons que la vitesse de convergence dans le cas où r=5 est plus grande par rapport au cas décrit par la figure 4.3 . Ce qui explique que la proportion des cluster heads qui optent pour la stratégie P augmente d'une manière prompt, car une récompense importante incite les cluster heads à adopter la stratégie P rapidement ;
- Comme nous voyons dans la figure 4.7 l'augmentation de **l** induit une vitesse de convergence encore plus grande que dans le cas où nous augmentons seulement la récompense. Ceci est dû au fait que les cluster heads sont plus encouragés à participer quand les risques liés aux pertes sont importants ;
- Pour $p_1(0)$=0.25, la figure 4.8 nous montre qu'une augmentation de c, malgré qu'elle est accompagnée d'une augmentation de r et l, entraîne une convergence vers la stratégie évolutionnairement stable p_1^*=0, contrairement au cas de la figure 4.3 où le réplicateur converge vers p_1^*=1. Ainsi, une récompense importante n'est pas toujours suffisante pour que tous les cluster heads finissent par choisir P. De même pour les pertes.

En guise de conclusion, nous pouvons dire que le réplicateur dynamique converge vers les deux stratégies évolutionnairement stables, p_1^*=0 et p_1^*=1. Ceci dépend de l'état initial et des paramètres du jeu. Les conditions de convergence vers l'une de ces deux stratégies sont alors résumées comme suit :

1. Si $p_1(0) > \frac{c}{r+l}$: le réplicateur dynamique converge vers la stratégie P ;
2. Si $p_1(0) < \frac{c}{r+l}$: le réplicateur dynamique converge vers la stratégie NP ;

3. Si $p_1(0)=\frac{c}{r+l}$: le réplicateur dynamique ne quitte pas son état intial au cours du temps car cet état est un état stationnaire.

Ces conditions confirment les résultats théoriques.
Alors, le taux initial au delà duquel le réplicateur dynamique converge vers la stratégie évolutionnairement stable $p_1^*=1$ décroît en augmentant la valeur de $r+l$ et en réduisant le c . Ainsi, la sécurité sera garantie même si la proportion initiale de cluster heads qui adopte la stratégie P n'est pas très importante.

4.5 Simulation et interprétation des résultats

Après avoir décrit notre modèle dans la section précédente, nous passons maintenant à son évaluation. La majorité des travaux sur l'évaluation des performances utilisent le principe de la simulation, vus les avantages qu'elle offre. En effet, la simulation constitue actuellement l'outil le plus pratique pour évaluer le comportement d'un système complexe, dont la formulation à l'aide de méthodes analytiques est difficile. Le principe consiste à modéliser la globalité du système étudié et à le simuler. Elle permet alors, d'observer le comportement complet des systèmes et leurs performances avant l'implantation en cas réels.

L'intérêt de la simulation est de pouvoir travailler sur des systèmes non disponibles. Par exemple, lors de l'étape de conception, il est beaucoup moins coûteux de réaliser une simulation préalable des alternatives envisagées. De plus la simulation est un moyen très souple d'étudier un problème. Cette technique permet des réexécutions de programmes avec changement de paramètres.

Il existe deux types de simulations :

- La simulation de systèmes continus : un système est modélisé sous forme d'équations différentielles qui régissent son évolution. Nous trouvons dans ce cas, les simulations des marchés économiques, les simulations écologiques et bien d'autres ;
- La simulation de systèmes à évènements discrets : la modélisation de tels systèmes correspond à des règles de succession d'événements. L'évolution du système dépend alors des événements déjà survenus et de l'ordre temporel de ceux-ci.

Dans notre cas, il s'agit de simulation à événements discrets. Notre choix de langage de développement s'est porté sur Matlab.

Nous présenterons dans cette section, l'architecture de notre simulateur, les différents modèles utilisés, ainsi que l'interprétation des résultats de la simulation effectuée.

4.5.1 Description du simulateur

Il existe plusieurs simulateurs de réseaux qui prennent en compte les différentes caractéristiques de ces derniers. Pour focaliser notre étude uniquement sur le comportement de notre modèle, nous avons utilisé un simulateur à évènements discrets

que nous avons développé et implémenté sous Matlab. Ce simulateur modélise un réseau ad hoc en utilisant notre proposition.

Ce simulateur permet alors de saisir les différents paramètres d'un réseau ad hoc et de notre modèle. Comme il nous permet également d'implémenter les procédures de formation et de gestion de ce réseau.

4.5.1.1 Paramètres du réseau

Les paramètres d'entrée du simulateur sont résumés dans le tableau suivant :

Variable	Sa définition	Type	Unité de mesure
(X, Y)	La couverture du réseau	(Réel, Réel)	(Mètre, Mètre)
N	Nombre de nœuds du réseau	Entier	—
R	Rayon de transmission d'un nœud	Réel	Mètre
$[v_{min}, v_{max}]$	Vitesse minimale et vitesse maximale d'un nœud	(Réel, Réel)	Mètre/Seconde
E-initiale	Vecteur contenant l'énergie initiale de chaque nœud	Réel	Joule
Seuil	Energie minimale pour qu'un nœud soit prioritaire à être élu cluster head	Réel	Joule
Taux-IPP	Taux initial de la population participant à la détection	Entier	—
T	Temps de simulation	Réel	Seconde
h	Le pas de simulation	Réel	Seconde

TABLE 4.2 – Variables d'entrée du simulateur

4.5.1.2 Modèle de mobilité

L'implémentation de ce modèle est une partie indispensable dans la simulation. Il s'agit alors de définir la loi du mouvement des différents nœuds du réseau. Le modèle utilisé est **Random Walk** [14], dans lequel un nœud mobile se déplace de son emplacement actuel vers un nouvel emplacement en choisissant aléatoirement une direction et une vitesse de déplacement. La nouvelle vitesse et la nouvelle direction sont choisies uniformément dans $[v_{min},\ v_{max}]$ et $[0,\ 2\pi]$, respectivement. Chaque mouvement est dans un intervalle de temps t constant, à la fin duquel une nouvelle direction et une nouvelle vitesse sont générées.

Si un nœud mobile qui se déplace en fonction de ce modèle atteint une limite de simulation, il "rebondit" sur les frontières de la simulation avec un angle déterminé par la future direction. Il continue son déplacement selon ce nouveau chemin.

Le modèle de mobilité Random Walk est un schéma de mobilité sans mémoire c'est à dire que la vitesse et la direction d'un nœud mobile sont indépendantes de son passé.

4.5.1.3 Mise à jour du niveau d'énergie

Afin de mettre à jour le niveau d'énergie des nœuds, nous avons pris en compte dans notre simulation la quantité consommée lors de l'émission et réception des messages. A un instant donné, les nœuds du réseau peuvent ne pas émettre ni recevoir de messages. Pour cela, une génération de messages a été faite suivant une loi de Bernoulli dans le but de désigner l'ensemble des nœuds concernés par la transmission où l'émission de messages. Nous avons également pris en considération l'énergie qu'un nœud consomme lors de l'activation de son IDS. La manière dont la mise à jour s'effectue sera présentée dans la section suivante.

4.5.1.4 Algorithme de clusterisation

Dans le but de partitionner le réseau en clusters, l'algorithme de clusterisation proposé a été implémenté. Ainsi, le nombre de nœuds de contrôle est déterminé qui est représenté par l'ensemble des cluster heads.

4.5.1.5 Génération d'attaques

Les nœuds attaqués seront choisis de manière aléatoire parmi l'ensemble total des nœuds et ceci suivant une loi de Bernoulli. Par conséquent, le nombre d'attaques auquel le réseau est exposé sera alors fixé.

4.5.1.6 Replicateur dynamique

Le réplicateur dynamique a été implémenté afin d'obtenir le nombre de cluster heads qui contribuent à la sécurité, en choisissant d'activer leurs IDS respectifs, à chaque instant de la simulation. Dans le cas où ces cluster heads ou l'un de leurs membres est sujet à une attaque, elle sera alors détectée. Le nombre total de détections sera donc déterminé.

Nous présenterons dans la figure 4.9, un schéma récapitulatif de notre application, comprenant les différentes étapes de sa conception.

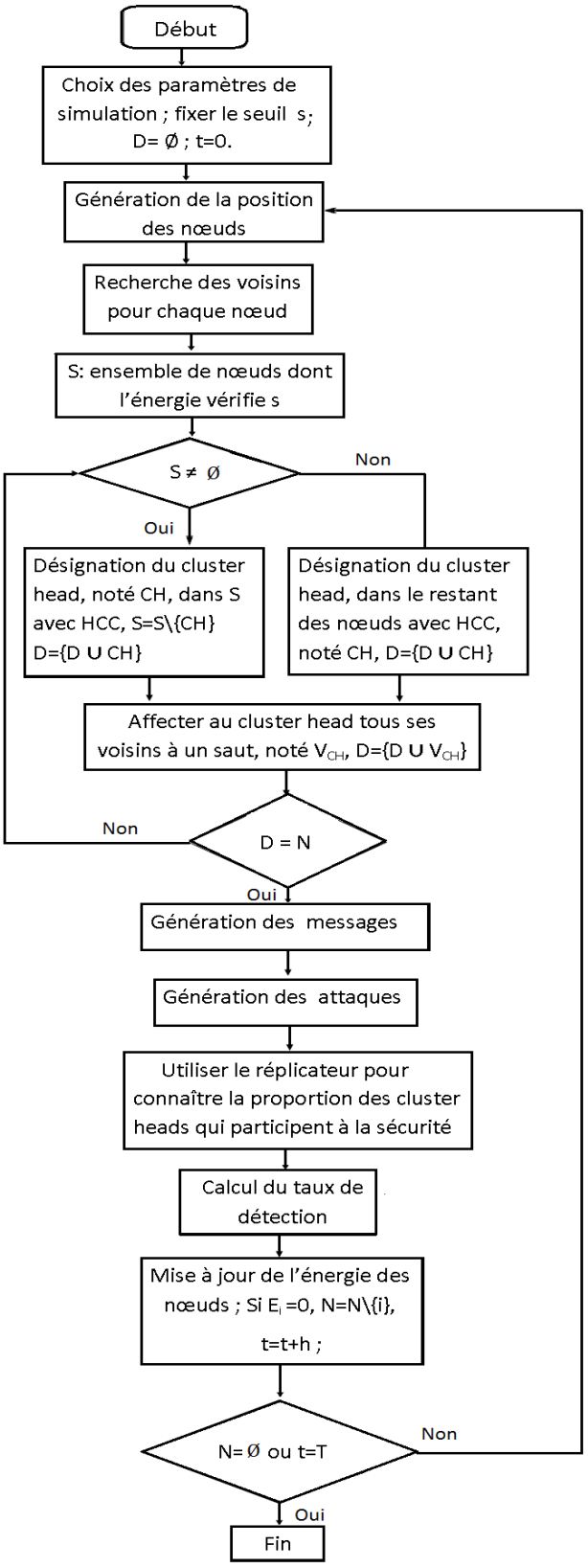


FIGURE 4.9 – Organigramme de l'application.

4.6 Paramètres de la simulation

Le réseau ad hoc implémenté est défini par sa couverture [2] et le nombre nœuds qu'il contient. Le réseau choisi est un réseau 1000*1000m comprenant N=20 nœuds. Chaque nœud du réseau est défini par :

- Sa position dans le réseau : la position est définie par les coordonnées cartésiennes du nœud (x, y) suivant les axes des abscisses X et des ordonnées Y.
- Un modèle de mobilité : la vitesse v $\in$ [0, 15m/s] [4] et la direction de mouvement $\theta \in [0, 2\pi]$;
- Un rayon de transmission : noté R, dont la valeur est fixée à 250m et qui est la même pour tous les noeuds du réseau ;
- Le type du noeud : un nœud peut être soit un cluster head, soit un nœud membre ;
- Un niveau d'énergie : initialement, les nœuds ont tous la même quantité d'énergie, fixée à 100 J [20], qui diminue dans les cas suivants :
  1. A chaque réception et transmission de paquets, une quantité choisie d'une manière aléatoire dans l'intervalle [5, 15J] sera retranchée,
  2. Quand l'IDS du nœud en question est activé, dans ce cas la quantité sera égale à 10J [4].

La période de simulation s'étale sur 150 secondes. Les nœuds changent de position chaque 15 secondes avec des directions et des vitesses différentes. Le seuil requis pour qu'un nœud devienne prioritaire à être élu cluster head est fixé à 25J. Après chaque changement de la topologie, l'algorithme de la clusterisation et la procédure du réplicateur dynamique seront exécutes, ainsi que la génération des attaques.

(X, Y)	1000*1000m
N	20
R	250m
$[v_{min}, v_{max}]$	[0, 15]
Einitial	100J
Seuil	25J
Taux-IPP	0.56
T	150 secondes
h	15 seconde

TABLE 4.3 – Paramètres de la simulation

4.7 Résultats de la simulation

La simulation qui dure 150 secondes, a été faite conformément aux paramètres mentionnés dans le tableau 4.3. Les métriques que nous avons choisies pour l'évaluation des performances du modèle présenté, sont essentiellement : le taux de détection

2. Taille en coordonnées X et Y.

des attaques, le nombre de nœuds inactifs [3] et le nombre de nœuds isolés [4] qui donne la probabilité de connectivité du réseau.

Les résultats affichés par l'éditeur ont permis de déterminer : le taux de détection des attaques générées, le nombre de nœuds inactifs et le nombre de nœuds isolés. Ce dernier nous donne la probabilité de connectivité du réseau. L'exécution du simulateur nous permet également d'obtenir les résultats des procédures implémentées durant la durée de simulation ainsi que la représentation graphique des dernières positions des nœuds comme le montre la figure suivante :

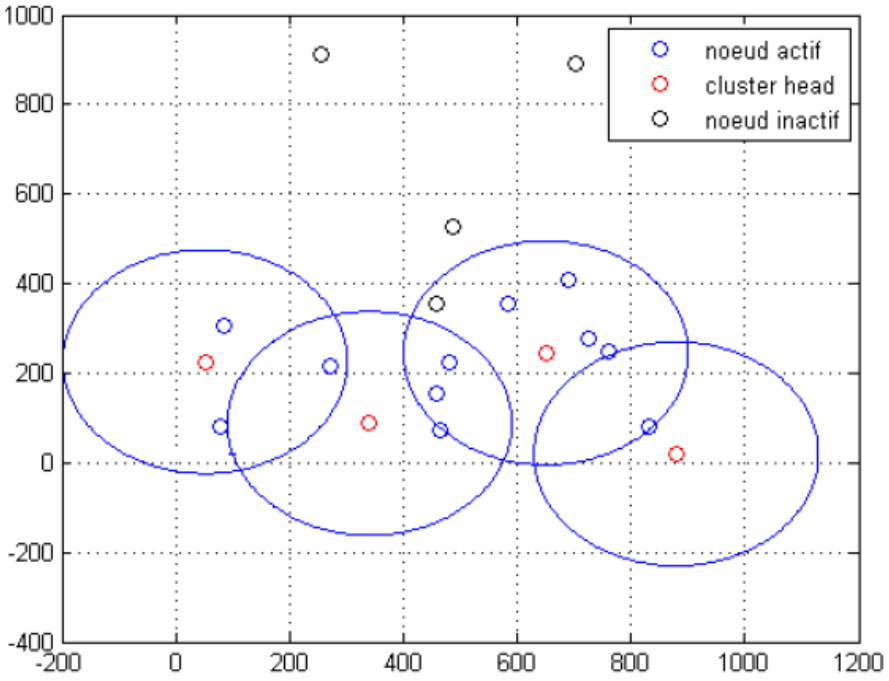


FIGURE 4.10 – Positions des nœuds.

La figure 4.10 illustre les positions des nœuds à la dernière itération, ainsi que les différents clusters. Les clusters sont représentés par les cercles bleus dont les nœuds de couleur rouge correspondent aux cluster heads et ceux de couleur bleu représentent les membres, quant aux nœuds inactifs ils sont représentés par la couleur noire. Ces derniers peuvent être considérés comme des nœuds isolés, malgré qu'ils se trouvent géographiquement dans un cluster ou plus, du fait qu'ils apparaissent invisibles aux autres nœuds et ne peuvent maintenir la connectivité du réseau.

Nous allons présenter maintenant un tableau récapitulatif des résultats de l'algorithme de clusterisation contenant : les différents instants où les nœuds changent de position et les clusters formés à chacun de ces instants.

3. Un nœud inactif est un nœud dont l'énergie est égale à 0.
4. Un nœud isolé est un nœud dont l'ensemble des voisins à un saut est nul.

t	Cluster heads	Membres du cluster	Energie des nœuds de 1 à 20
0	7	1, 5, 6, 8, 10, 13, 20.	[100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100]
	11	8, 13, 15, 19.	
	2	3, 12, 18, 20.	
	17	1, 4, 9.	
	14	15.	
	16		
15	7	1, 5, 6, 8, 10, 13, 20.	[93.64, 90.00, 86.87, 100.00, 87.61, 90.17, 90.00, 100.00, 100.00, 88.75, 84.36, 87.05, 100.00, 90.00, 100.00, 90.00, 76.24, 94.46, 100.00, 88.35]
	2	3, 5, 12, 18, 20.	
	17	1, 4, 9.	
	19	11, 15, 16, 18.	
	14		
30	7	1, 4, 5, 8, 10, 11, 13, 16, 17, 20.	[93.64, 66.01, 80.52, 100.00, 82.34, 82.75, 72.56, 100.00, 91.94, 82.91, 84.36, 74.55, 87.60, 71.84, 93.57, 69.00, 66.24, 86.33, 79.58, 82.73]
	18	1, 16, 20.	
	6	5, 10, 13.	
	15	14, 19.	
	12	2, 3.	
	9	4, 17.	
45	8	4, 7, 9, 10, 11, 13, 17.	[80.40, 66.01, 66.54, 100.00 76.26, 72.75, 53.53, 94.18, 71.17, 74.52, 71.05, 64.55, 87.60, 59.21, 83.55, 55.79, 56.87, 76.33, 74.47, 82.73]
	20	1, 7, 16, 18 .	
	12	2, 3, 5, 6.	
	15	5, 6, 13.	
	19	13.	
	14		
60	10	4, 8, 9, 11, 13, 15, 17.	[80.40, 51.47, 66,.54, 100.00 76.26, 61.38, 53.53, 71.83, 59.84, 63.82, 71.05, 45.42, 77.21, 35.57, 73.55, 50.48, 45.67, 76.33, 64.47, 72.73]
	16	1, 7, 8, 17, 18,20.	
	3	1, 2, 18, 20.	
	12	1, 5, 6.	
	19	13.	
	14		
75	4	9, 10, 11, 13, 15, 17.	[67.00, 51.47, 56.54, 100.00 64.39, 51.59, 42.69, 57.25, 47.61, 43.07, 59.35, 28.75, 77.21, 25.57, 73.55, 30.86, 47.67, 63.41, 42.75, 66.79]
	1	2, 3, 6, 7, 18, 20.	
	19	11, 13, 15, 17.	
	16	7, 8, 15, 20.	
	12	5, 6.	
	14		

90	4	9, 10, 11, 13, 15, 16, 17.	[57.00, 46.13, 56.54, 90.00
	3	31, 2, 6, 7, 18, 20.	64.39, 51.59, 42.69, 57.25,
	8	10, 15.	47.61, 29.48, 50.44, 18.75,
	5	12.	67.58, 9.55, 67.65, 20.86
	19		47.67, 63.41, 32.75, 66.79],
	14		
105	17	4, 8, 9, 10, 11, 13, 15, 16.	[57.00, 46.13, 36.25, 69.39,
	1	2, 3, 6, 7, 20.	54.39, 44.29, 42.69, 47.25,
	18	2.	37.11, 29.48, 50.44, 3.89,
	5	12.	67.58, 0.00, 67.65, 20.86,
	19		47.67, 55.57, 22.75, 66.79]
120	15	4, 7, 8, 13, 17, 20.	[38.80, 39.33, 36.25, 69.39
	10	4, 8, 9, 11, 13, 17.	31.55, 44.29, 42.69, 47.25,
	6	1, 2, 3, 7, 18.	28.51, 29.48, 50.44, 3.89,
	5		54.54, 0.00, 67.65, 20.86,
	16	3, 12, 20.	35.67, 36.42, 12.75, 66.79]
	19		
135	4	8, 10, 11, 13, 15, 17, 20.	[38.80, 34.25, 36.25, 69.39,
	7	1, 2, 6, 18, 20.	14.20, 34.29, 36.95, 37.84,
	3		19.61, 19.48, 43.27, 0.00,
	9	10, 11, 13, 17	54.54, 0.00, 50.24, 0.00
	19		35.67, 27.69, 2.75, 59.28]
	5		
	9	2.	
150	15	3, 4, 9, 11, 13, 17, 20.	[29.66, 34.25, 26.25, 47.38,
	1	4, 7, 8, 20.	0.00, 19.47, 19.97, 23.28,
	2	6, 7, 18.	0.95, 5.55, 29.51, 0.00,
	10	9	54.54, 0.00, 50.24, 0.00
			30.33, 13.27, 0.00, 48.70]

TABLE 4.4 – Résultats de la clusterisation

Le tableau 4.4 nous permet de voir que les cluster heads et leurs membres peuvent changer au cours du temps. Nous remarquons que l'énergie d'un nœud diminue à chaque instant, mais avec des quantités différentes et ceci pour deux raisons : son changement de statut, passer de cluster head vers membre ou vice-versa, et le changement du nombre de messages qu'il reçoit ou émet.

Notons S l'ensemble des nœuds qui dépassent le seuil fixé par l'algorithme de clusterisation, 25J. Jusqu'à l'instant 75, cet ensemble est égale à l'ensemble de tous les nœuds du réseau, S=20. Ce qui fait que l'élection s'effectue en prenant en compte juste le degré de chaque nœud. A partir de l'instant 90, le cardinal de l'ensemble S commence à se réduire. Ainsi, l'élection des cluster heads se fait sur un nombre restreint de nœuds.

A partir de l'instant 105, l'énergie de certains nœuds commence à devenir nulle ce qui diminue la connectivité du réseau car nous considérons les nœuds inactifs comme étant des nœuds isolés et qui ne peuvent appartenir à aucun cluster.

4.7.1 Impact de Taux-IPP sur le taux de détection des attaques

Dans cette section, nous allons évaluer l'impact du taux initial de cluster heads activant leur IDS sur le taux de détection moyen. Comme nous l'avons déjà mentionné, le réplicateur dynamique a été implémenté dans le simulateur et ceci nous permet de tirer la proportion de cluster heads ayant adopté la stratégie P à chaque instant de la simulation où nous générons des attaques. Ainsi, en disposant de l'ensemble des cluster heads dont la stratégie choisie est P et de l'ensemble des nœuds attaqués, nous pourrions obtenir le taux de détection à chaque fois qu'il ait génération d'attaques. Afin d'observer comment évolue le taux de détection en fonction du taux initial de cluster heads choisissant la stratégie P, nous avons fixé des valeurs pour le Taux-IPP et à chacune d'elle le simulateur nous renvoie le taux de détection moyen associé, qui correspond à la moyenne des résultats obtenus en effectuant 10 simulations. Les paramètres du jeu ont été aussi fixés : r=3, l=2 et c=1. Les résultats obtenus sont illustrés dans la figure 4.11 :

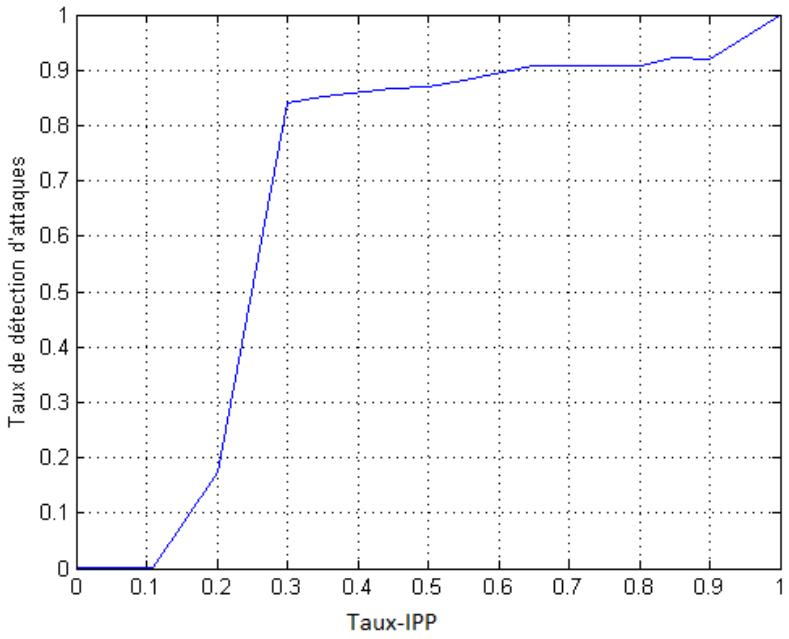


FIGURE 4.11 – Impact de Taux-IPP sur le taux de détection des attaques.

D'après la figure 4.11, nous observons que le taux de détection d'attaques n'est pas trop important pour un taux de participation initial qui ne dépasse pas 0.2. Au delà de cette valeur, nous remarquons une augmentation considérable de celui-ci jusqu'à atteindre 1. Ceci est dû au fait que pour les mêmes paramètres du jeu : r=3, l=2 et c=1, le réplicateur dynamique converge vers la stratégie évolutionnairement stable p_1^*=0 dans le cas, où la proportion initiale de la population adoptant la stratégie P ne dépasse pas 0.2 et converge vers p_1^*=1 dans le cas contraire, chose que nous avons constatée lors de la présentation des résultats de son implementation. Par conséquent, si la proportion initiale est en dessous de 0.2, toutes les attaques générées à partir du moment où le réplicateur dynamique atteint la stratégie évolutionnairement stable p_1^*=0 ne seront pas détectées, car tous les cluster heads choisissent de ne pas participer, ce qui explique le niveau bas de détection pour des taux initiaux qui ne dépassent pas 0.2. Et dans le cas, où la proportion initiale de cluster heads choisissant P dépasse 0.2, le taux de détection associé est élevé car de tels états finissent toujours par atteindre la stratégie évolutionnairement stable p_1^*=1. Ainsi, à partir de cet instant, toutes les attaques générées seront détectées, car tous les cluster heads choisissent de participer et c'est ce qui augmente le taux de détection total pour ces états.

Pour conclure, le taux de détection est une fonction croissante du taux initial de la population optant pour P.

4.7.2 Evaluation de la clusterisation

Le modèle proposé, basé sur la clusterisation, a été conçu dans l'objectif de conserver l'énergie des nœuds tout en maintenant la sécurité du réseau et ceci en activant que les IDS des cluster heads. Pour mettre en valeur les apports de ce mécanisme, nous nous sommes proposé de faire une étude comparative entre notre modèle, avec clusterisation, et un autre où il est supposé que tous les nœuds du réseau constituent des nœuds de contrôle. Nous nous sommes alors focalisés sur les critères suivants : le nombre moyen de nœuds inactifs et le nombre moyen de nœuds isolés.

Avant de passer à la comparaison, nous allons présenter l'interface graphique de notre modèle que nous avons créé à l'aide de Matlab. Cette interface prend en valeur d'entrées les paramètres de simulation déjà donnés et nous renvoi des résultats correspondants aux : nombre moyens de nœuds isolés, nombre moyen de nœuds inactifs et la probabilité moyenne de connectivité. Nous aurons également, comme sortie, deux graphes qui représentent le nombre de nœuds actifs en fonction du temps et la représentation des nœuds dans le réseau.

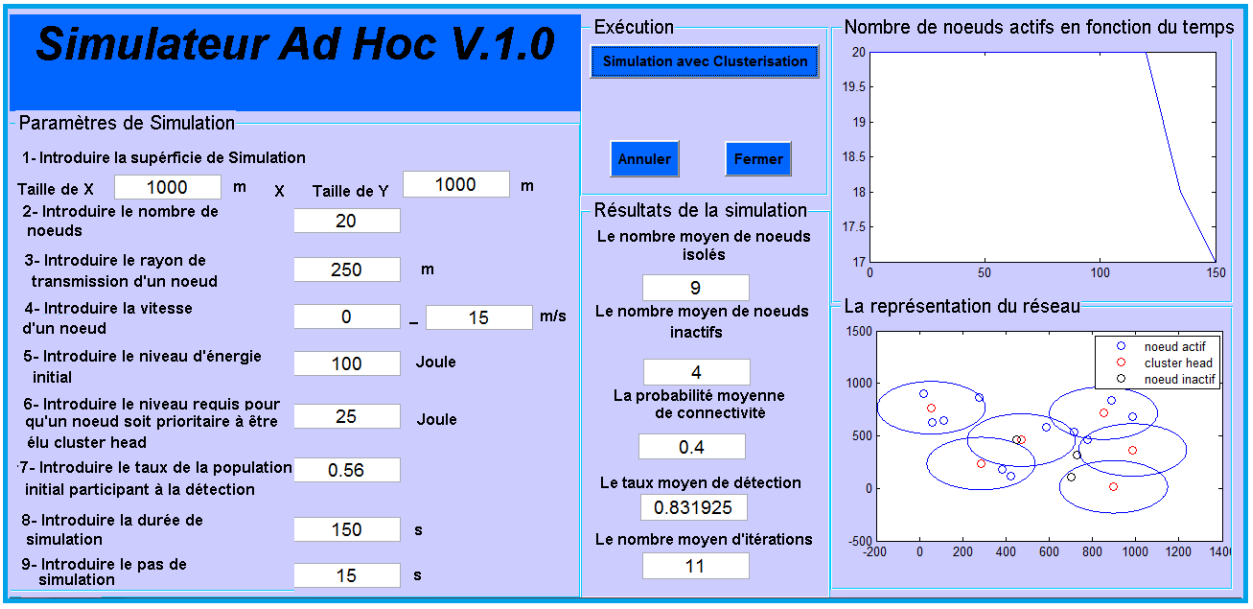


FIGURE 4.12 – Interface graphique du simulateur

Afin d'effectuer la comparaison, nous avons intégré dans notre interface le modèle où nous avons supposé que tous les nœuds du réseau constituaient des nœuds de contrôle :

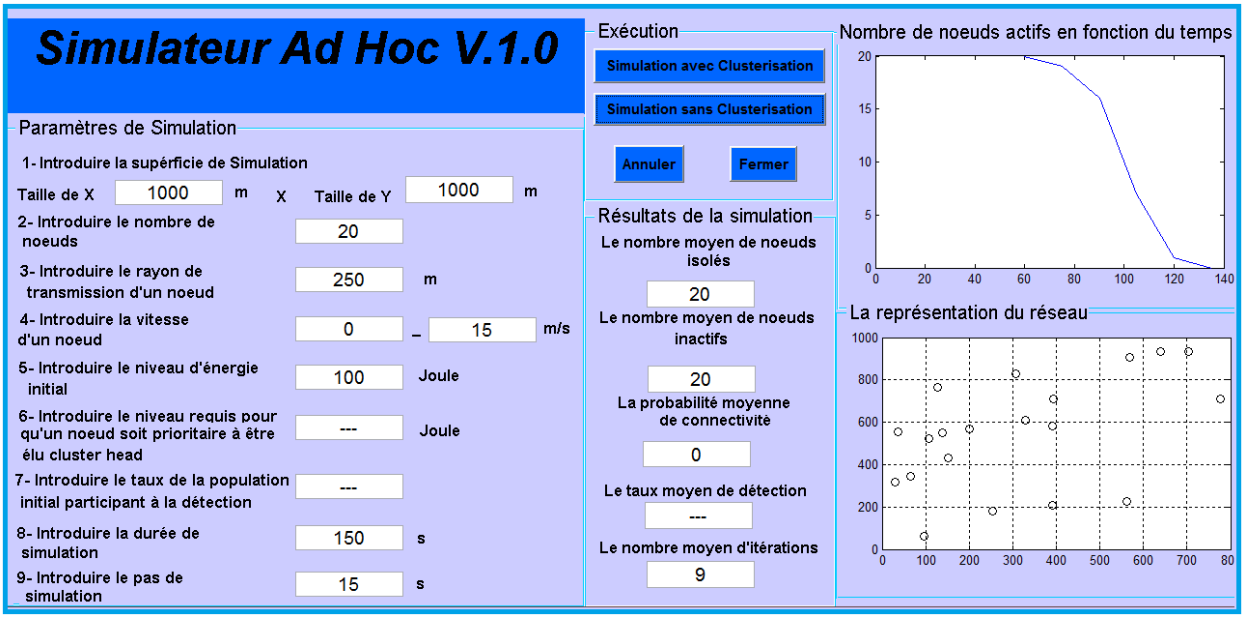


FIGURE 4.13 – Interface graphique du simulateur

Pour les mêmes paramètres de simulation, le nombre de nœuds actifs en fonction du temps pour les deux modèles, avec clusterisation et sans clusterisation est représenté par la figure suivante :

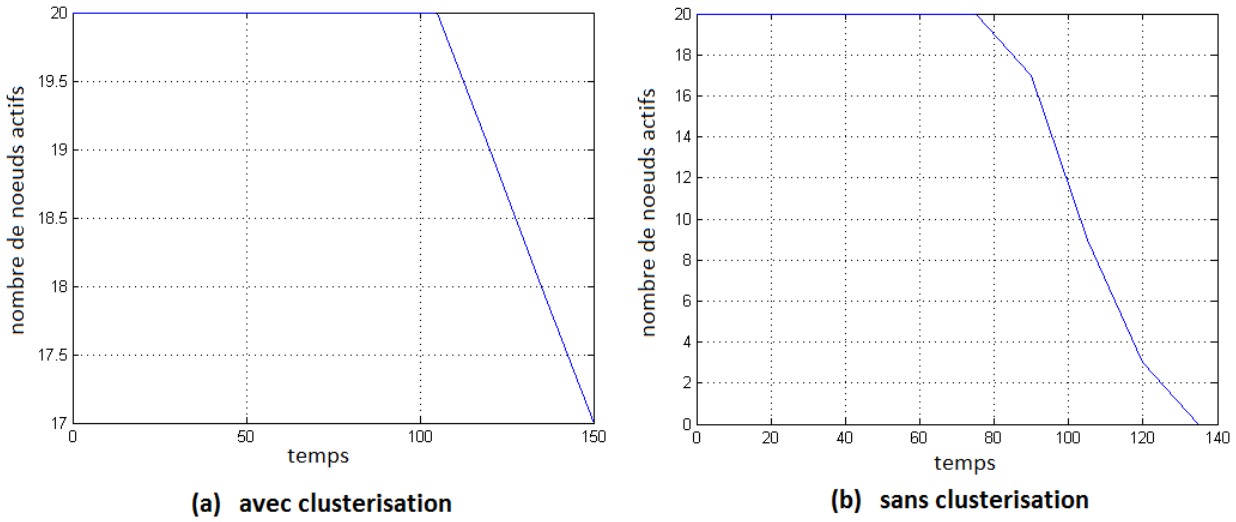


FIGURE 4.14 – Nombre de nœuds actifs en fonction du temps

Pour les deux modèles, nous remarquons qu'à l'instant 0 tous les nœuds du réseau sont actifs en raison de leur niveau d'énergie initial assez important. Ce nombre reste inchangé jusqu'à l'instant où l'un des nœuds devient inactif. Cet instant correspond alors à 135 pour la figure (a) et à 90 pour la figure (b) où nous remarquons que 3 nœuds deviennent inactifs simultanément. Nous remarquons également sur la figure

(b), qu'à partir du moment où le premier nœud devient inactif le nombre de nœuds actifs décroît considérablement jusqu'à devenir 0 à l'instant 135. Par conséquent, tous les nœuds deviennent inactifs avant même que le temps de simulation ne soit écoulé. En revanche, le nombre de nœuds actifs associés au modèle avec clusterisation décroît très lentement et à la fin de la simulation nous ne trouvons que 3 nœuds inactifs.

En somme, les résultats concernant le nombre de nœuds actifs donnés par le modèle avec clusterisation sont nettement meilleurs que ceux obtenus par le modèle sans clusterisation. Cette différence montre bien comment la clusterisation et l'activation des IDS uniquement sur les cluster heads influe sur le nombre de nœuds actifs dans un réseau.

Quant au nombre moyen de nœuds isolés et la probabilité moyenne de connectivité du réseau durant la simulation, ils sont donnés dans le tableau suivant :

	Avec clusterisation	Sans clusterisation
Nombre moyen de nœuds isolés	9	20
Probabilité moyenne de connectivité	0.4	0

TABLE 4.5 – Connectivité du réseau

Nous constatons à partir du tableau qu'un réseau partitionné en clusters est plus connecté qu'un réseau non clusterisé, ceci est dû au fait que les nœuds inactifs sont considérés comme étant des nœuds isolés car ils ne peuvent participer à aucune fonctionnalité du réseau. Comme le nombre de nœuds inactifs, à la fin de la simulation, dans le modèle sans clusterisation est supérieur à celui avec clusterisation cela confirme les résultats illustrés dans le tableau 4.5.

4.8 Conclusion

Dans ce chapitre, nous avons traité le problème de la sécurité dans les réseaux ad hoc en proposant un modèle basé sur la théorie des jeux évolutionnaires.

Nous avons, par le biais d'une clusterisation, optimiser les ressources en énergie des nœuds et mis en avant l'aspect évolutif du réseau pour pouvoir suivre l'interaction existant entre les cluster heads du réseau dans leur participation au processus de sécurité.

Nous avons mis en évidence la convergence du réplicateur dynamique, qui décrit l'évolution de la proportion des cluster heads jouant la stratégie "Protéger", vers l'ESS. Nous avons également conçu un simulateur d'un réseau ad hoc sous Matlab dans lequel nous avons intégré un algorithme de clusterisation et le réplicateur dynamique. D'après les simulations réalisées à évènements discrets, nous avons pu constater que le taux de détection était une fonction croissante du taux initial de

la population optant pour P. D'après aussi ces simulations, la technique de clusterisation est efficace pour réduire le nombre de nœuds inactifs et donc le nombre de nœuds isolés, ce qui rend le réseau plus connecté tout en maintenant sa sécurité.

Conclusion Générale

Les réseaux ad hoc sont un nouveau type de réseaux qui suscite l'intérêt des praticiens des réseaux et des chercheurs. Ils rencontrent cependant deux contraintes majeures qui peuvent entraver leur développement : le problème de la sécurité et la contrainte d'énergie.

Les problèmes de sécurité dans les réseaux ad hoc sont difficiles à résoudre car ces réseaux sont de nature dynamique et sans infrastructure préexistante. Ces caractéristiques empêchent l'utilisation des solutions de sécurité déjà existantes, pour cela l'introduction de la théorie des jeux a été nécessaire afin de cerner l'interaction entre les différents agents du réseau. Quant à la contrainte de l'énergie, les nœuds disposent d'une énergie limitée par la capacité de leur batterie qui est difficilement rechargeable en cours de déploiement.

Dans notre travail, nous avons proposé un modèle pour la sécurité des réseaux ad hoc tout en considérant leur capacité limitée en énergie. Ce modèle est composé de deux étapes : la première consiste en une procédure de clusterisation dont le but est de conserver l'énergie des nœuds et la seconde repose sur une modélisation du comportement des cluster heads en moyennant la théorie des jeux évolutionnaires.

Ce rapport s'est consacré en premier lieu à la présentation des réseaux et particulièrement les réseaux ad hoc ainsi que les impératifs de sécurité. Ensuite nous avons introduit la théorie des jeux en définissant les notions de base qui lui sont liées, ses concepts de solution et ses caractéristiques principales pour mieux comprendre le raisonnement qu'elle apporte. Nous avons par la suite présenté certains travaux qui ont été faits dans le contexte où la théorie des jeux trouve sa place pour être appliquée comme étant l'outil fondamental pour la résolution des problèmes de sécurité des réseaux ad hoc.

La dernière partie a été consacrée à la présentation de notre modèle en détaillant les différentes étapes le constituant. Nous avons commencé par la présentation de l'algorithme de clusterisation proposé basé sur l'algorithme HCC et le niveau d'énergie des nœuds, ensuite nous avons décrit le jeu évolutionnaire se déroulant entre les cluster heads dont l'ensemble de stratégies est de contribuer ou non à la sécurité du réseau. Nous avons par la suite trouvé les ESS du jeu, calculé le réplicateur dynamique et donné les conditions de sa convergence vers l'une des ESS. Ces conditions ont été confirmées lors de l'implementation du réplicateur dynamique sous Matlab. Nous avons également décrit notre simulateur de réseaux ad hoc, créé sous Matlab,

où nous avons intégré le réplicateur dynamique afin de voire l'évolution du taux de détection en fonction de la proportion initiale de cluster heads contribuant à la sécurité. Dans le but de voire l'apport de la clusterisation, nous avons effectué une comparaison entre notre modèle et un modèle sans clusterisation, en prenant comme métrique l'évolution du nombre de nœuds inactifs dans le temps, d'après les résultats obtenus nous avons pu constater que la clusterisation diminue le nombre de nœuds inactifs tout en maintenant le réseau sécurisé.

En guise de perspectives, nous pouvons utiliser un modèle de consommation d'énergie plus spécifique, qui distingue entre l'énergie consommée lors de l'émission et celle consommée lors de la réception d'un message où la distance entre l'émetteur et le récepteur est prise en compte. Ceci refléterait d'avantage la réalité.

Bibliographie

[1] R. Agarwal, M. Motwani, "Survey of Clustering Algorithms for MANET," International Journal on Computer Science and Engineering, Vol. 1, No. 2, pp. 98-104, 2009.

[2] P. Albers, O. Camp, B. Jouga, L. Mé, J-M. Percher, R. Puttini, "Un Système de Détection d'Intrusions Distribué pour Réseaux Ad Hoc," RSTI-TSI. Sécurité Informatique, No. 23, pp. 391-420, 2004.

[3] D.J. Baker, A. Ephremides, J. E. Wieselthier. "A Design Doncept for Reliable Mobile Radio Networks with Frequency Hopping Signaling". In Proceedings of the IEEE, Vol. 75, No. 1, pp. 56-73, January 1987.

[4] P. Bhattacharya, M. Debbabi, N.Mohammed, H. Otrok, L. Wang, "Mechanism Design-Based Secure Leader Election Model for Intrusion Detection in MANET," IEEE Transaction on Dependable and Secure Computing, Vol 99, ISSN 1545-5971, 2009.

[5] M. Chatterjee, S. K. Das, D. Turgut, "WCA : A Weighted Clustering Algorithm for Mobile Ad Hoc Networks," Cluster Computing, No. 5, pp. 193-204, 2002.

[6] G. Chen, I. Stojmenović. "Clustering and Routing in Mobile Wireless Networks". Technical Report TR-99-05, University of Ottawa, June 1999.

[7] C. Comaniciu, Y. Liu, H. Man, "A Bayesian Game Approach for Intrusion Detection in Wireless Ad Hoc Networks," Proceedings of GameNets (Workshop on Game Theory for Networks), Pise, Italie, pp. 1-12, Octobre, 2006.

[8] F. Djemili Tolba, "Conservation d'Energie et Gestion de la Mobilité dans les Réseaux Ad Hoc," Thèse de Doctorat, Département d'Informatique, Université Franche-Comté, 2007.

[9] M.Gerla, J. Tzu-Chieh Tsai, "Multicluster Mobile, Multimedia Radio Network," ACM/Baltzer Journal of Wireless Networks, Vol. 1, No. 3, pp. 225-265, July 12, 1995.

[10] E. Habamungu Kalume, "Etude et Amélioration des Performances d'un Réseau MAN," Mémoire de Fin d'Etudes, Institut Supérieur de Statistique et de Nouvelles Technologies du Congo, 2010.

[11] L. B. Jonker, P. D. Taylor, "Evolutionary Stable Strategies and Game Dynamics," Mathematical Biosciences, Elsevier, No. 40, pp. 145-156, 1978.

[12] C. A. Kamhoua, K. Makki, N. Pissinou, "Game Theoretic Modeling and Evolution of Trust in Autonomous Multi-Hop Networks Application to Network Security and Privacy," IEEE Communications Society, pp. 1-6, june 2011.

[13] A. I. Konaté, "Etude et Conception de l'Architecture du Système de Détection d'Intrusions RIDAN," Rapport de Fin d'Etudes, Ecole Supérieure des Communication de Tunis, 2006.

[14] P. Manzoni, M. Sanchez, "A Java Based Simulator for Ad hoc networks," Future Generation Computer Systems, Vol. 17, No. 5, pp. 573-583, 2001.

[15] P. Michiardi, "Application de la Théorie des Jeux et de l'Evolution dans le Cadre d'Observabilité Imparfaite," Actes du Symposium SSTIC06, 2006.

[16] H. Moulin, "Théorie des Jeux pour l'Economie et la Politique," Hermann, 1981.

[17] J. Nash, "Equilibrium Point in N-Person Games," Proccedings of the National Academy of Sciences, No. 36, pp. 48-49, 1950.

[18] J. M. Park, A. Patcha, "A Game Theoretic Formulation for Intrusion Detection in Mobile Ad Hoc Networks," International Journal of Network Security, Vol. 2, No. 2, pp. 131-137, Mar. 2006.

[19] A. Rachedi, "Contributions à la Sécurité dans les Réseaux Mobiles Ad Hoc," Thèse de Doctorat, Université d'Avignon et des pays de Vaucluse, 2008.

[20] M. Saad, "Contrôle Intelligent de Flux Capable de s'Adapter à l'Etat d'un MANET," Thèse de Doctorat, Université Mouloud Mammeri de Tizi Ouzou.

[21] J. M. Smith, "Evolution and the Theory of Games," Cambridge University Press, 1982.

[22] H. Sun, H. Wei, "Using Bayesian Game Model for Intrusion Detection in Wireless Ad Hoc Networks," Int. J. Communications, Network and System Sciences, Vol. 3, No. 7, pp. 602-607, 2010.

[23] G. Vache-Marconato, "Evaluation Quantitative de la Sécurité Informatique : Approche par les Vulnérabilités," Thèse de Doctorat, Univerdité de Toulouse, 2009.

[24] K. Wei Lye, J-M. Wing, "Game Strategies in Network Security," Int J Inf Secur No. 4, pp. 71-86, 2005.

[25] J.W. Weibull, "Evolutionnary Game Theory," Cambridge, MA : The M.I.T. Press, 1995.

[26] M. Yildizoglu, "Introduction à la Théorie des Jeux," Dunod, 2003.

Résumé

Les réseaux ad hoc constituent une technologie émergente offrant à leurs utilisateurs de nombreux avantages en termes de coût et de facilité d'utilisation. Cependant, ils sont soumis à une multitude de challenges, en particulier la mobilité des nœuds, le problème de ressources limitées, comme l'énergie, mais aussi leur vulnérabilité en terme de sécurité qui suscite, ces dernières années, la réaction de nombreuses recherches. En effet, les nœuds dans un réseau ad hoc mobile doivent contrecarrer diverses attaques et actions malveillantes. Plusieurs mécanismes de sécurité exigent la coopération de plusieurs nœuds afin de défendre le réseau contre ces attaques.

Dans ce travail, nous avons proposé une nouvelle perspective permettant de répondre à ces besoins. Cette solution s'est constituée en premier lieu en une clusterisation du réseau afin de mieux gérer les ressources des nœuds le composant. Ensuite nous avons modéliser le comportement stratégique des cluster heads sous forme de jeu évolutionnaire.

Mots clés : *réseaux ad hoc, sécurité informatique, clusterisation, IDS, théorie des jeux, théorie des jeux évolutionnaires, réplicateur dynamique, ESS.*

Abstract

Wireless ad hoc network constitute emerging technology offering to their users many advantages in terms of cost and ease of use. However, they are subject to a multitude of challenges, particularly the mobility of nodes, the problem of limited resources, such as energy, but also their vulnerability in terms of security which arose, in recent years, the reaction of much research. Indeed, the nodes in a mobile ad hoc network must thwart various attacks and malicious actions. Several security mechanisms require the cooperation of several nodes in order to defend the network against these attacks.

In this paper, we propose a new perspective to address these needs. This solution is consisted primarily in clustering of the network for better managing the resources of its nodes. then we have modeled the strategic behavior of cluster heads as evolutionary game.

Keywords : *ad hoc networks, network security, clustering, IDS, game theory, evolutionary game theory, ESS, replicator dynamic.*