

CHAPITRE 3

CONSTRUCTION D'UNE DORSALE VIRTUELLE

3.1 Introduction

Dans les réseaux sans fil *ad hoc*, les terminaux communiquent entre eux sans l'intermédiaire d'une entité physique. La communication a tendance à être inefficace en termes de traitement et d'utilisation des ressources dans le réseau. L'étude d'une structure partielle ou dorsale virtuelle attire l'attention de plusieurs chercheurs dans le but de réduire *l'overhead* de la communication. Mais la structure d'une dorsale virtuelle est très vulnérable à cause de plusieurs facteurs comme la mobilité des nœuds, l'instabilité des liens, etc. Ainsi, un mécanisme efficace capable de construire et de maintenir la structure de la dorsale est indispensable tout en considérant la stabilité et la couverture des nœuds.

L'absence d'une infrastructure physique augmente non seulement le coût de communication mais fait aussi face à un problème sévère connu sous le nom d'orage d'inondation (*Broadcast Storm Problem*) [Tseng *et al.* (2002)]. Ce problème est engendré par le processus de diffusion des messages de contrôle dans les protocoles de routage qui utilisent la technique d'inondation. Tseng *et al.* ont démontré qu'une rediffusion après la première inondation ne peut uniquement atteindre que 41 % d'espace à couvrir, et ce pourcentage décroît de 5 % à chaque itération. Ils ont démontré également que deux récepteurs d'un même message diffusé se contentent de rediffuser le message avec une probabilité de 0,59 ce qui résulte en une très faible probabilité de l'existence des hôtes sans contention. En conclusion, l'inondation engendre des problèmes tels que : la redondance, la contention et la collision des messages à la réception.

Dans ce chapitre, nous allons détailler notre algorithme proposé pour la construction de la dorsale au moment de l'établissement. Ensuite, une analyse probabiliste sera présentée pour évaluer notre algorithme. Enfin, nous comparerons notre algorithme avec d'autres approches proposées dans la littérature pour construire une dorsale dans un réseau.

3.2 Description détaillée du problème

La minimisation du nombre de messages échangés est requise dans un réseau *ad hoc* où la bande passante est une ressource précieuse, et un réseau trop chargé peut provoquer des pertes de paquets considérables. Minimiser le nombre de messages revient à minimiser le nombre de nœuds qui vont être désignés pour retransmettre les messages de contrôle. L'ensemble de ces nœuds sera alors appelé une dorsale virtuelle. La construction et la maintenance de cette dorsale imposent l'ajout d'un *overhead* de contrôle qui est proportionnel à la taille de la dorsale. Par conséquent, la taille de l'ensemble des nœuds qui forment la dorsale doit être la plus petite possible. De plus, le rôle de cette dorsale nécessite que les nœuds qui forment cette dorsale soient connectés. De ce fait, la détermination de l'ensemble de domination de taille minimale (*Minimum Connected Dominating Set*, MCDS) dans un graphe présente une solution adéquate pour construire la dorsale.

Dans la théorie des graphes, l'ensemble de MCDS dans un graphe est l'ensemble des nœuds tels que chaque nœud est soit élément de l'ensemble MCDS, soit il possède un voisin dans cet ensemble. Rappelons dans ce contexte qu'un voisin est le nœud se trouvant dans la zone de couverture du transmetteur.

Dans ce chapitre, nous supposons que tous les nœuds ont le même rayon de couverture et par la suite nous pouvons modéliser le réseau en utilisant un graphe de disque unitaire, *Unit-Disk Graph* UDG. Toutefois, la détermination de l'ensemble MCDS dans un graphe UDG est un problème *NP-Complexe* [Das et Bharghavan (1997)].

La connaissance de la position de chaque nœud dans le réseau nécessite l'utilisation d'un système de localisation tel que le GPS. Dans le cas où un tel système n'est pas disponible, il existe des outils de localisation utilisés dans les réseaux à capteurs (systèmes statiques) tels que Metricom Ricochet et Nokia Rooftop. Il existe aussi des algorithmes efficaces tels que VCap (*Virtual Coordinate assignment protocol*) présenté dans [Caruso *et al.* (2001)] ou APS (*Ad hoc Positioning System*) proposé par Niculescu *et al.* dans [Niculescu *et al.* (2003)]. D'autres travaux de recherche ont proposé des algorithmes pour déterminer la position des nœuds dans les réseaux *ad hoc* en se basant sur la puissance reçue et en supposant qu'on peut déduire la distance entre deux nœuds connaissant cette puissance [Capkun *et al.* (2001)], [Youssef *et al.* (2005)] et [Dimitri et Dudek (2006)]. Ces algorithmes utilisent généralement les méthodes de trilatération, triangulation ou multilatération pour calculer la position relative d'un nœud par rapport à un autre. Pour déterminer l'ensemble MCDS dans un graphe, nous supposons que la position des nœuds est connue au moment de l'établissement du réseau. En effet, un de ces algorithmes peut être alors utilisé pour déterminer la position de chaque nœud dans le réseau.

Le mouvement des nœuds dans les réseaux *ad hoc* rend la construction d'une dorsale plus difficile. Nous présumons qu'une connaissance appropriée de la topologie du réseau au moment de son établissement est acquise. La topologie du réseau peut être établie suite à l'information sur le nombre des terminaux et leur position. Il est alors faisable de déterminer la dorsale en considérant un système statique. Les nœuds identifiés pour être éléments de cette dorsale formeront une zone de service pour tout le réseau, et prendront en charge des fonctionnalités de localisation des mobiles, de diffusion des informations de topologie, etc. Cette structure doit s'adapter au déplacement des nœuds dans le réseau *ad hoc* de nature dynamique, une procédure sera exécutée par chaque nœud afin de maintenir la dorsale connectée. Dans le chapitre 5, nous allons détailler la procédure de maintenance qui sera exécutée par chaque nœud.

Une dorsale virtuelle permet de réduire le nombre de terminaux qui vont s'occuper de la tâche de diffusion des messages dans tout le réseau. En effet, seuls les membres qui forment cette dorsale auront à retransmettre les messages de contrôle reçus. Cette méthode réduit aussi la consommation de la bande passante dans le réseau, et par la suite l'énergie consommée par les terminaux. Nous avons intérêt à ce que cette dorsale comprenne un minimum de terminaux possible; la taille de cette dorsale doit être aussi la plus petite possible.

L'utilisation d'une dorsale virtuelle s'avère très avantageuse :

- elle permet de limiter l'inondation, puisque les messages de contrôle peuvent être acheminés de façon fiable et économique dans tous les réseaux; seuls les nœuds qui forment la dorsale auront à relayer les messages de contrôle,
- elle peut être utilisée pour collecter l'information sur la topologie pour fin de routage et fournir une route de secours en cas de besoin,
- elle joue un rôle primordial dans le cas des réseaux hybrides, des réseaux *ad hoc* connecté au réseau Internet. En effet, un terminal membre de la dorsale peut être désigné comme un point d'accès (*Access Point, AP*),
- elle est utile pour les applications multipoints, chaque nœud de la dorsale peut jouer le rôle d'un relais multipoint chargé de répliquer les paquets multipoint.

3.3 Hypothèses, notations et définitions

Un réseau *ad hoc* est généralement représenté par un graphe $G = (V, E)$, avec V l'ensemble des nœuds et $E \in V^2$ l'ensemble des arcs donnant les communications directes possibles : (u, v) appartient à E si et seulement si u peut envoyer directement un message à v (on dit alors que v est voisin de u). Les couples appartenant à E dépendent de la position des nœuds et de leur portée de communication. Nous prenons l'hypothèse que tous les nœuds ont la même portée R . Soit $d(u, v)$, la distance entre les nœuds u et v . L'ensemble E est défini alors comme suit :

$$E = \{(u, v) \in V^2 \mid d(u, v) \leq R\} \quad (1)$$

Ce graphe est connu sous le nom de graphe unitaire UDG normalisé par R , avec R rayon de transmission ou rayon de recouvrement. Dans ce graphe, $G = (V, E)$, nous définissons $N = |V|$ comme le nombre de nœuds dans le réseau. Le degré d'un nœud u , Δ_u , représente le nombre des nœuds voisins de u , défini par $\{v \mid (u, v) \in E\}$. Le degré moyen d'un réseau est alors égal à :

$$\Delta = \frac{\sum_{u \in V} \Delta_u}{N} \quad (2)$$

Un nœud v est appelé un voisin dominant d'un autre nœud u s'il existe un lien direct. Un sous-ensemble de nœuds est dominant si chaque nœud n'appartenant pas à ce sous-ensemble possède au moins un voisin dominant dans ce sous-ensemble. On désigne par Δ_{\max} degré maximal du graphe $\Delta_{\max} = \max\{\Delta_u \mid u \in E\}$.

Déf. 1 : Un graphe $G = (V, A)$ est appelé connexe si et seulement si $\forall (u, v) \in V, \exists$ un chemin entre u et v .

Déf. 2 : Un ensemble $S \subset V$ est appelé dominant si chaque nœud de $G, \notin S$, possède au moins un voisin dans S .

$$\forall u \in V, \exists v \in S / \exists (u, v) \in E \mid (u, v) \in E \}$$

Déf. 3 : Basé sur le concept de domination, chaque nœud non-dominant possède un voisin dominant.

Un ensemble dominant est un sous-ensemble de nœuds du graphe tel que chaque nœud du graphe est soit dans l'ensemble dominant, soit qu'il possède un voisin dans l'ensemble dominant. Par exemple, dans la figure 18-a, les nœuds noirs forment un ensemble de

domination de taille minimale. Chaque nœud dans le réseau n'appartenant pas à cet ensemble, est connecté à au moins un nœud de cet ensemble de domination. La figure 18-b illustre l'ensemble de domination connexe pour le même exemple. L'ensemble des nœuds en gris seront alors ajoutés à l'ensemble des nœuds en noirs pour former la dorsale.

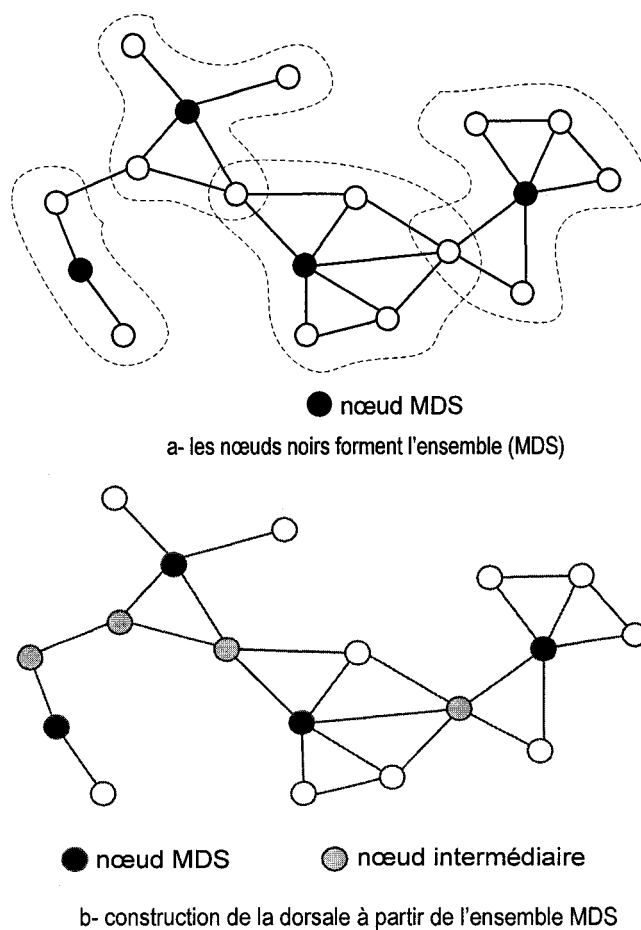


Figure 18 Construction d'une dorsale dans un graphe

Rappelons que la détermination de l'ensemble de domination connexe de taille minimale dans un graphe est un problème *NP-Complexe* [Das et Bharghavan (1997)]. Il existe plusieurs heuristiques pour la détermination de l'ensemble de domination connexe.

Quelques-unes essayent de minimiser la taille de cet ensemble, d'autres non. Comme nous l'avons présenté dans le Chapitre 2, toutes ces heuristiques sont basées sur des approches combinatoires basées sur la théorie de coloration dans les graphes ou sur les processus de marquage. De plus, elles commencent par déterminer l'ensemble de domination connexe CDS puis elles essaient de le minimiser. Notre approche, différentes des autres, consiste à trouver, dans un premier temps, l'ensemble de domination de taille minimale (MDS) dans un réseau puis déterminer l'ensemble des nœuds intermédiaires pour avoir un ensemble MDS connexe. Nous allons vérifier au moyen des simulations que notre approche fournit une solution minimale de la taille de la dorsale comparée à d'autres algorithmes proposés dans la littérature.

3.4 Description de l'algorithme MDS-based

Dans cette section, nous allons présenter un nouveau algorithme (CDS-based) pour déterminer une approximation de l'ensemble MCDS dans un graphe UDG. Nous avons décomposé ce problème en deux étapes [Mnif *et al.* (2005a)]. Notre approche consiste à déterminer dans un premier temps l'ensemble de domination de taille minimale (MDS) en utilisant une formulation en programmation linéaire. Une fois cet ensemble déterminé, une deuxième étape consiste à trouver l'arbre de recouvrement d'un graphe réduit formé par l'ensemble MDS. Nous obtiendrons ainsi l'ensemble des nœuds qui vont former la dorsale. Le diagramme de la figure 19 illustre les étapes de résolution pour trouver l'approximation de l'ensemble MCDS.

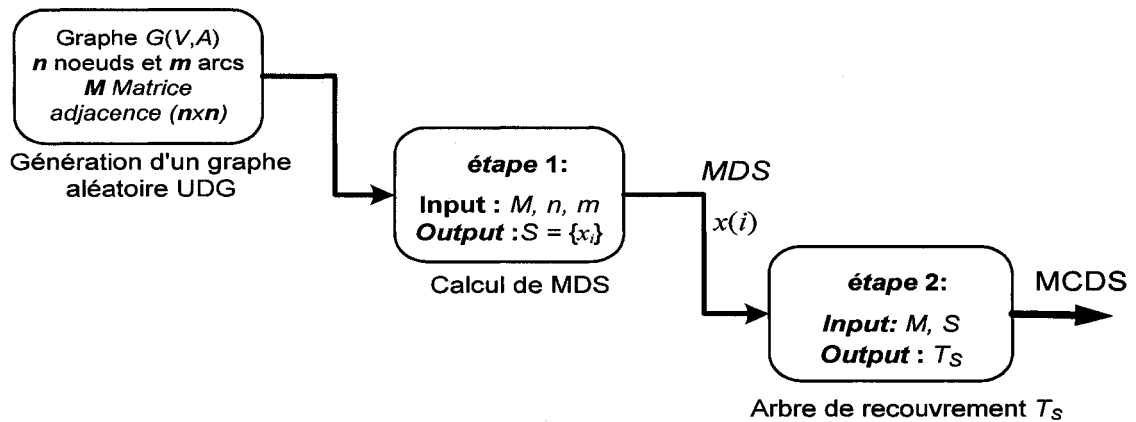


Figure 19 Étapes de résolution

Théorème 1

Soit $G = (V, E)$ comme défini dans la section 3.3, nous allons démontrer que :

$S \subset V$ est un ensemble de domination connexe de G si et seulement si un arbre de recouvrement T de G tel que $\mathcal{V}S$ est un sous-ensemble des nœuds terminaux de T .

Preuve

Soit $S \subset V$ un ensemble de domination connexe de G . Soit T_S un arbre de recouvrement minimum de sous graphe de G induit par les nœuds de S . Pour chaque nœud $w \in \mathcal{V}S$, choisissons un nœud $u_w \in S \cap N(w)$, avec $N(v) = \{u \in V \mid (u,v) \in E\}$. Il est évident de voir que $T = T_S \cup \{u_w \mid w \in \mathcal{V}S\}$ est un arbre de recouvrement de G et $\mathcal{V}S$ est un sous ensemble des nœuds externes de T (les nœuds ayant un degré égal à 1). Inversement si T est un arbre de recouvrement de G et U est un sous ensemble des nœuds externes de T , alors pour tout nœud de U est relié à au moins un nœud de $\mathcal{V}U$ et le sous graphe de G induit par les nœuds de $\mathcal{V}U$ est connexe. Alors $\mathcal{V}U$ est un ensemble de domination connexe de G .

3.4.1 Détermination de l'ensemble de domination MDS (étape 1)

La première étape consiste à minimiser l'ensemble des nœuds dominants dans un graphe donné. Pour cela, nous définissons une variable de décision x_i tel que :

$$x_i = \begin{cases} 1 & \text{si le nœud } i \text{ appartient à l'ensemble de domination} \\ 0 & \text{sinon} \end{cases}$$

La fonction objectif cherche à minimiser la taille de l'ensemble de domination S , sous la contrainte que chaque nœud dans le graphe G possède au moins un nœud voisin dans cet ensemble de nœuds dominants S .

$$(P) \quad \min \sum_{i \in V} x_i \quad (3)$$

La contrainte de domination s'écrit :

$$X + M \times X \geq 1 \quad (4)$$

$$\text{Avec } x_i \in \{0,1\} \quad (5)$$

avec $X = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$ représente le vecteur de décision,

et M est une matrice d'adjacence ($n \times n$) de G , $M = \begin{bmatrix} m_{11} & \cdots & m_{1n} \\ \vdots & m_{ij} & \vdots \\ m_{n1} & \cdots & m_{nn} \end{bmatrix}$, $m_{ij} = 1$ si et seulement

si le nœud i est connecté au nœud j , et $m_{ij} = 0$ autrement.

Avec cette formulation, la solution en x_i est fournie dans un temps de calcul de l'ordre de $O(n)$. En se référant à [Megiddo (1987)], un problème en programmation linéaire avec d variables et n contraintes, le temps de sa résolution est de l'ordre de $O(n)$, avec n peut tendre vers l'infini.

À cette étape, nous n'avons aucune garantie sur la connectivité de la solution fournie par la résolution de ce problème. Pour cela et pour s'assurer que la solution finale soit connexe, nous allons déterminer l'ensemble des nœuds intermédiaires pour que l'ensemble MDS soit connexe.

3.4.2 Détermination de l'ensemble MDS Connexe (étape 2)

La résolution du problème (P) nous donne l'ensemble de domination de taille minimale (MDS) qui n'est pas nécessairement connexe. Afin de trouver les nœuds intermédiaires, nous allons déterminer l'arbre de recouvrement d'un graphe réduit construit à partir de l'ensemble MDS.

Cette étape suivante consiste à :

- construire un graphe réduit G' de façon à ce que :
 - les sommets de ce graphe sont formés par l'ensemble MDS,
 - deux nœuds sont reliés par le plus court chemin en nombre de sauts calculé à partir du graphe initial,
 - le poids sur un arc représente le nombre de sauts entre deux nœuds.
- déterminer l'arbre de recouvrement de taille minimale, T , dans le graphe G' ,
- extraire les nœuds intermédiaires en remplaçant les arcs choisis par les chemins réels dans le graphe original.

Pour déterminer le plus court chemin entre deux nœuds, nous avons utilisé l'algorithme de *Dijkstra*. L'algorithme *Dijkstra* peut s'exécuter dans un temps de l'ordre de $O(m + n \log n)$ en utilisant l'implémentation de tas de *Fibonacci* [Ahuja et al. (1993)], avec n et m représentant respectivement le nombre de nœuds et le nombre d'arcs dans le graphe G .

Soit le graphe réduit $G'(V', E')$ formé à partir de l'ensemble MDS. Soit n' et m' représentent respectivement le nombre de nœuds et le nombre d'arcs dans le graphe

réduit G' . La formulation pour déterminer T_S l'arbre de recouvrement minimum MST (*Minimum Spanning Tree*) dans un graphe s'écrit alors :

La variable binaire y_{ij} indique si l'arc (i, j) fait partie de l'arbre de recouvrement minimum, T . En d'autres termes :

$$y_{ij} = \begin{cases} 1 & \text{si les noeuds } i \text{ et } j \text{ appartiennent à l'arbre de recouvrement minimum } T_S \\ 0 & \text{sinon} \end{cases}$$

$$\min \sum_{(i,j) \in V'} c_{ij} y_{ij} \quad (6)$$

Sous les contraintes :

$$\sum_{(i,j) \in T_S} y_{ij} = n_T - 1 \quad (7)$$

$$\sum_{(i,j) \in \pi(R)} y_{ij} \leq |R| - 1 \quad \forall R \subseteq T_S, \quad (8)$$

$$\text{Avec } y_{ij} \in \{0,1\} \quad (9)$$

Dans cette formulation, le coefficient c_{ij} désigne le coût sur l'arc (i, j) reliant le nœud i et le nœud j . Dans notre cas, le coût sur un arc représente le nombre de sauts entre les nœuds i et j .

La contrainte (7) est une contrainte de cardinalité ; l'arbre comprend exactement $n_T - 1$ arcs. Avec $n_T = |T_S|$ représente le nombre de liens dans le sous graphe engendré par T_S . La contrainte (8) est appelée *packing constraint* ; l'ensemble d'arcs choisis ne doit pas contenir de cycle (si la solution choisie contient un cycle et R est l'ensemble de nœuds dans ce cycle, la solution viole alors cette contrainte) [Ahuja *et al.* (1993)]. T_S est l'arbre de recouvrement minimum pour S qui contient la solution finale de l'ensemble des nœuds dominant connexe de taille minimale. $\pi(R)$ dénote l'ensemble des arcs dans le sous graphe de S induit par l'ensemble des nœuds R [i.e., $\pi(R)$ est un ensemble d'arcs de

E' avec des points finaux dans R]. Notons que, comme fonction du nombre de nœuds dans le réseau, ce modèle contient un nombre exponentiel de contraintes. Néanmoins, ce problème peut être résolu de façon efficace en utilisant les algorithmes de *Prim* et *Kruskal* [Cormen *et al.* (1994)]. Le temps d'exécution pour les algorithmes de *Prim* et *Kruskal* est de l'ordre de $O(m' + n' \log n')$. Dans notre cas, n' est la taille de l'ensemble MDS. Il existe aussi un algorithme distribué : l'algorithme de *Sollin* est une version hybride de les algorithmes de *Kruskal* et *Prim*. Le temps d'exécution de cet algorithme est de l'ordre de $O(m' \cdot \log n')$. Cet algorithme performe mieux que l'algorithme de *Prim* pour les graphes clairsemés (*sparse graphs*) [Ahuja *et al.* (1993)].

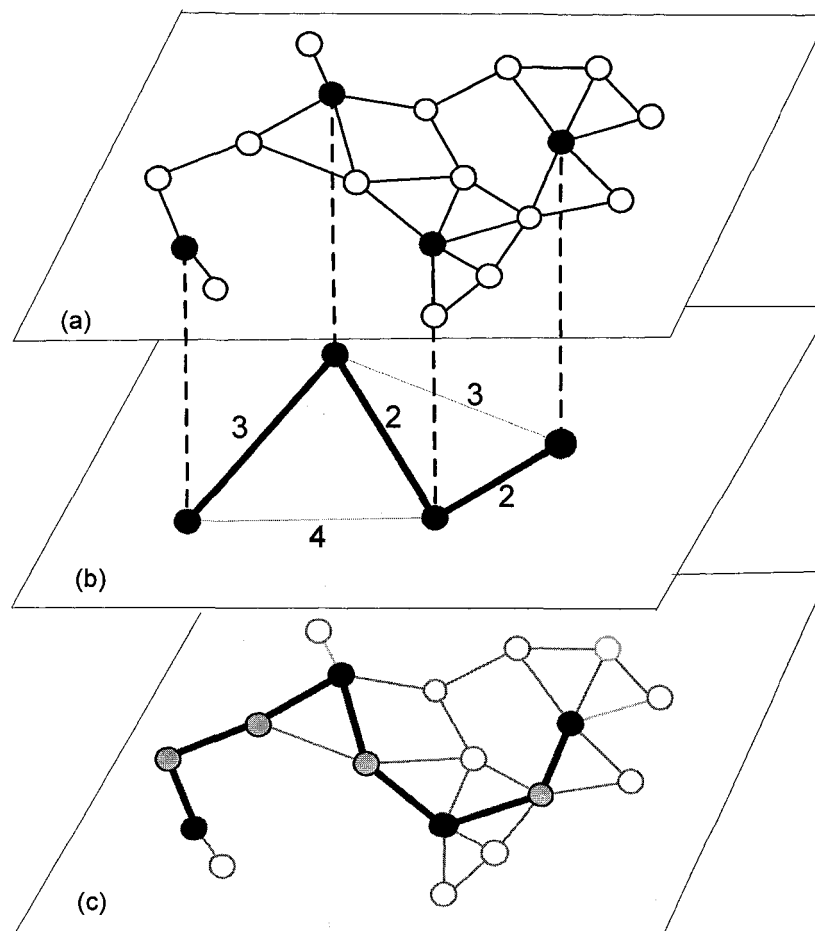


Figure 20 Détermination de la dorsale à partir de l'ensemble MDS

Reprenons l'exemple de la figure 18, la première étape de notre algorithme a permis de déterminer l'ensemble MDS (les quatre nœuds en noirs), comme illustré sur la figure 20-a. Nous construisons le graphe réduit composé de quatre nœuds et cinq arcs, le poids sur chaque arc représente le nombre de saut du plus court chemin entre deux nœuds. Nous déterminons ainsi l'arbre de recouvrement de taille minimale pour ce graphe réduit, figure 20-b. Nous extrairons les nœuds intermédiaires (les nœuds en gris) en effectuant la correspondance entre les arcs de l'arbre de recouvrement et les chemins dans le graphe original, figure 20-c. La dorsale est alors l'ensemble des nœuds noirs et gris et qui représente la dorsale pour cet exemple.

3.4.3 Analyse de performance

Dans cette section, une étude analytique sera présentée afin de vérifier que la taille de la dorsale trouvée en utilisant l'algorithme CDS-based est bien la taille minimale estimée. Pour ce faire, nous allons déterminer la probabilité p_{MCDS} qu'un nœud soit un élément de l'ensemble MCDS [Mnif *et al.* (2006)]. Connaissant la taille du réseau N , la taille estimée de l'ensemble MCDS sera alors :

$$N_{MCDS} = p_{MCDS} N \quad (10)$$

Désignons par $\rho = \frac{N}{S}$ la densité du réseau, avec N qui est le nombre total de nœuds et S la surface totale. Un nœud P ayant une portée, rayon de couverture égal à R , le nombre de nœuds N_1 qui peut être situé dans cette zone de centre P et de rayon R est :

$$N_1 = \rho S_1 \quad \text{avec } S_1 = \pi R^2$$

$$\text{alors } N_1 = \rho \pi R^2$$

Si les nœuds sont placés dans le plan suivant la distribution de Poisson¹³, alors la probabilité d'avoir k nœuds dans la surface S_1 est :

$$p_1(k, N_1) = \frac{N_1^k}{k!} e^{-N_1}, \text{ avec } N_1 = \rho S_1 \quad (11)$$

Un nœud est élément de l'ensemble MCDS si et seulement si l'une des deux conditions suivantes est vraie :

p_2 : le nœud est un élément de l'ensemble MDS (étape 1)

p_3 : le nœud est un nœud intermédiaire (étape 2)

La probabilité pour qu'un nœud soit l'élément de l'ensemble MCDS est la probabilité qu'un nœud soit un nœud de l'ensemble MDS, p_2 , ou bien un nœud intermédiaire, p_3 , sans être un nœud de l'ensemble MDS. Ceci est exprimé par l'équation suivante :

$$P_{MCDS} = p_2 + (1-p_2)p_3 \quad (12)$$

Dans une surface donnée et pour k nœuds, la probabilité pour un nœud d'être choisi parmi ses k nœuds voisins est donnée par $\frac{1}{k+1}$, alors :

$$p_2 = \sum_{k=1}^{\infty} p_1(k, N_1) \frac{1}{k+1} \text{ avec } p_1(k, N_1) = \frac{N_1^k}{k!} e^{-N_1}$$

¹³ la distribution de Poisson est caractérisée par le fait qu'elle est sans mémoire, purement aléatoire, elle est souvent utilisée dans ce contexte.

$$\begin{aligned}
p_2 &= \sum_{k=1}^{\infty} \frac{N_1^k}{k!} e^{-N_1} \frac{1}{k+1} \\
&= \frac{1}{N_1} \sum_{k=1}^{\infty} \frac{N_1^{k+1}}{(k+1)!} e^{-N_1} \\
&= \frac{1}{N_1} \sum_{k=2}^{\infty} \frac{N_1^k}{k!} e^{-N_1} \quad \text{or } \sum_{k=0}^{\infty} \frac{N_1^k}{k!} = e^{N_1} \\
&= \frac{1}{N_1} (e^{N_1} - 1 - N_1) e^{-N_1} \\
p_2 &= \frac{1 - e^{-N_1} (N_1 + 1)}{N_1} \tag{13}
\end{aligned}$$

Dans l'expression de p_2 , la somme commence à partir de $k = 1$ et non pas $k = 0$ parce que nous avons éliminé le cas où le nœud ne possède pas de voisin (aucun nœud ne se trouve dans S_I).

Pour évaluer la probabilité que le nœud soit un nœud intermédiaire p_3 , considérons la figure 21 où les deux nœuds I et J sont deux nœuds éléments de l'ensemble MDS et qui ne sont pas connectés. Supposons que les nœuds I et J sont à deux sauts, il existe alors un nœud K qui va être le nœud intermédiaire (nœud déterminé par l'étape 2 de l'algorithme). K doit se situer dans la zone grise $S(x)$ définie par l'intersection des deux cercles de centres I et J . Dans le cas général, I et J sont situés à n -sauts, les nœuds intermédiaires sont déterminés en utilisant le même principe.

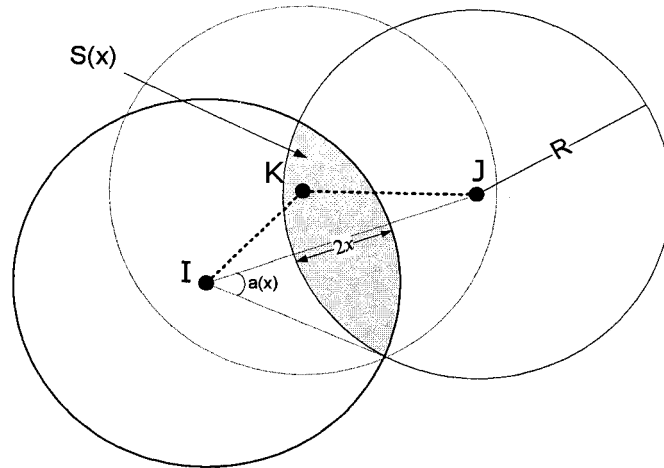


Figure 21 Le nœud K doit se situer dans la région S pour qu'il soit un nœud intermédiaire pour I et J .

Sur la figure 21, la surface $S(x)$ dépend de la distance qui sépare les nœuds I et J ; $d(I, J)$. En d'autres termes, S est fonction de la distance x , avec $x = R - \frac{d(I, J)}{2}$ pour $R < d(I, J) < 2R$.

$S(x)$ est donnée par [Spiegel et Liu (2000)] :

$$S(x) = 2R^2 a(x)$$

avec $a(x)$ est l'angle que fait l'axe (I, J) avec le point d'intersection des deux cercles de centres I et J (figure 21), $a(x)$ est donné par :

$$a(x) = \cos^{-1}\left(\frac{x}{R}\right) - \frac{x}{R} \sqrt{1 - \left(\frac{x}{R}\right)^2}$$

Désignons par $N(x)$ le nombre de nœuds situés dans la surface $S(x)$, nous avons alors :

$$N(x) = \rho S(x)$$

Et par la suite, la probabilité que le nœud soit un nœud intermédiaire est :

$$p_3 = \sum_{k=1}^{\infty} p_1(k, N(x)) \frac{1}{k+1}$$

avec $p_1(k, N_1) = \frac{N_1^k}{k!} e^{-N_1}$

En utilisant le même raisonnement dans le calcul de p_2 et le fait que $\sum_{k=0}^{\infty} \frac{N_1^k}{k!} = e^{N_1}$, nous obtiendrons alors :

$$p_3 = \frac{1 - e^{-N(x)}}{N(x)} - e^{-N(x)} \quad \text{pour } R < d(I, J) < 2R \quad (14)$$

Dans le cas où nous avons $d(I, J) = 2R$, le nœud K peut être alors considéré comme un nœud de l'ensemble MDS. Donc, ce cas particulier est inclus dans le calcul de p_2 . Vérifions ceci mathématiquement; si $d(I, J) = 2R$ alors l'intersection des deux cercles est un point et par la suite $S(x)=0$ et $N(x)=0$ ce qui donne $p_1=0$ et $p_3=0$.

Dans le cas où nous avons $d(I, J) = R$, les nœuds I et J peuvent communiquer directement et ce cas fait partie déjà de p_2 .

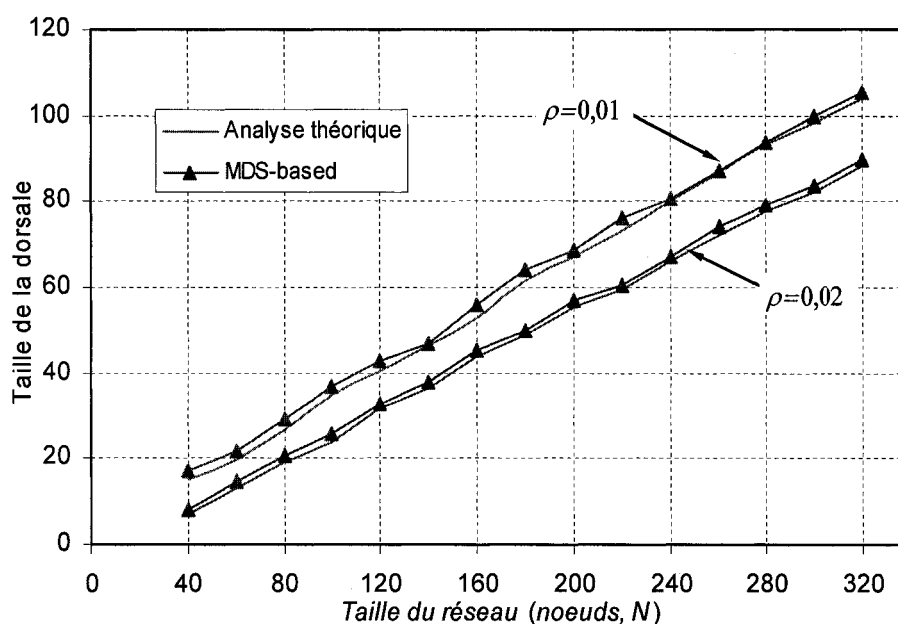


Figure 22 Taille de la dorsale en fonction de la taille du réseau

En utilisant MATLAB, nous allons comparer la taille de la dorsale, dans un graphe UDG donné, calculée par l'algorithme proposé (MDS-based) par rapport à celle donnée par l'analyse théorique (équation 8). Une première comparaison consiste à faire varier la taille du réseau. Les nœuds sont placés dans un plan 2D suivant une distribution exponentielle. La portée de chaque terminal est de $R = 20m$. Pour chaque valeur de N , nous effectuons dix simulations pour prendre une valeur moyenne. La figure 22 illustre les résultats de simulation pour $40 \leq N \leq 320$ pour deux valeurs de densité soit $\rho = 0.01$ et $\rho = 0.02$. La taille de la dorsale croît quasi-linéairement avec la taille du réseau. Les résultats de simulation sont légèrement supérieurs à ceux de l'analyse. L'écart relatif moyen entre les valeurs données par l'algorithme (MDS-based) et celles données par l'analyse est de l'ordre de 4 %.

Une deuxième simulation consiste à comparer la taille de l'ensemble MCDS en variant la portée de transmission, le rayon R , et de comparer les résultats trouvés par notre algorithme (MDS-based) et l'analyse (équation 8). Pour un nombre de nœuds fixe ($N =$

200) placés exponentiellement dans une surface ($100 \times 100 \text{ m}^2$), nous faisons varier le rayon de transmission. Pour chaque valeur de R , nous effectuons dix simulations pour prendre une valeur moyenne. La figure 23 illustre les résultats trouvés par l'algorithme MDS-based et l'analyse pour $10m \leq R \leq 100m$. La taille de l'ensemble MCDS décroît rapidement quand la portée augmente et ceci est expliqué par le fait que le nœud dominant couvre plus de voisins. Les résultats de simulations données par l'algorithme MDS-based sont très proches de ceux retrouvés par l'analyse.

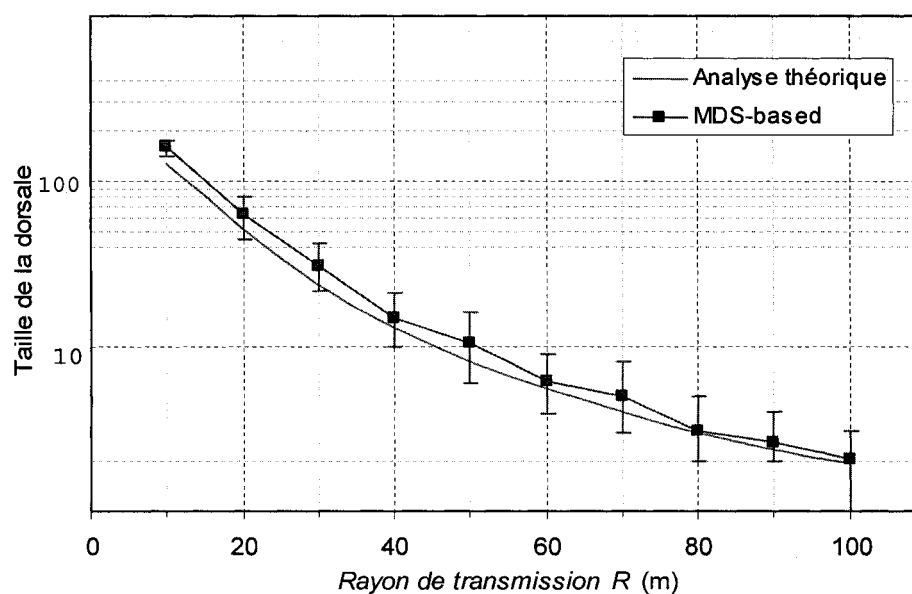


Figure 23 Taille de dorsale en fonction du rayon de transmission R

3.4.4 Comparaison avec d'autres algorithmes (WCDS, CDS-based et BCDS)

Dans cette section, nous allons comparer la solution fournie par notre algorithme (MDS-based) avec celle trouvée en utilisant les algorithmes décrits dans le chapitre 2 soit WCDS, CDS-based et BCDS dans ses deux versions. Plus la taille de la dorsale est petite, plus le résultat est meilleur. Rappelons que ces deux approches commencent par trouver un ensemble de domination connexe (CDS) dans le réseau puis elles essaient de

la minimiser. Alors que notre algorithme MDS-based détermine, dans une première étape, l'ensemble de domination de taille minimale et par la suite nous cherchons à déterminer l'ensemble MDS connexe.

Nous considérons des graphes dont la taille varie de 40 à 320 nœuds. Tous les nœuds possèdent le même rayon de transmission. Les nœuds sont générés et placés aléatoirement dans un plan carré. La taille de ce plan varie en fonction du nombre des nœuds pour avoir une densité constante (même degré moyen) pour toute les simulations. Afin de simuler la structure des réseaux *ad hoc*, deux nœuds seront connectés si la distance qui les sépare est inférieure à $R = 30 m$. Nous effectuons un prétraitement sur le graphe généré afin de s'assurer qu'il est bien connexe.

Pour chaque valeur de N , une trentaine de simulations ont été exécutées pour prendre une valeur moyenne de la taille de la dorsale. La figure 24 illustre les caractéristiques des graphes générés dans nos simulations. Les résultats démontrent que la procédure utilisée fournit un degré moyen constant avec une variance faible (≤ 0.2). Cette procédure est adéquate et fournit des graphes peu denses et connexes. Le nombre d'arcs croît de façon proportionnelle en fonction de la taille du graphe (nombre de nœuds).

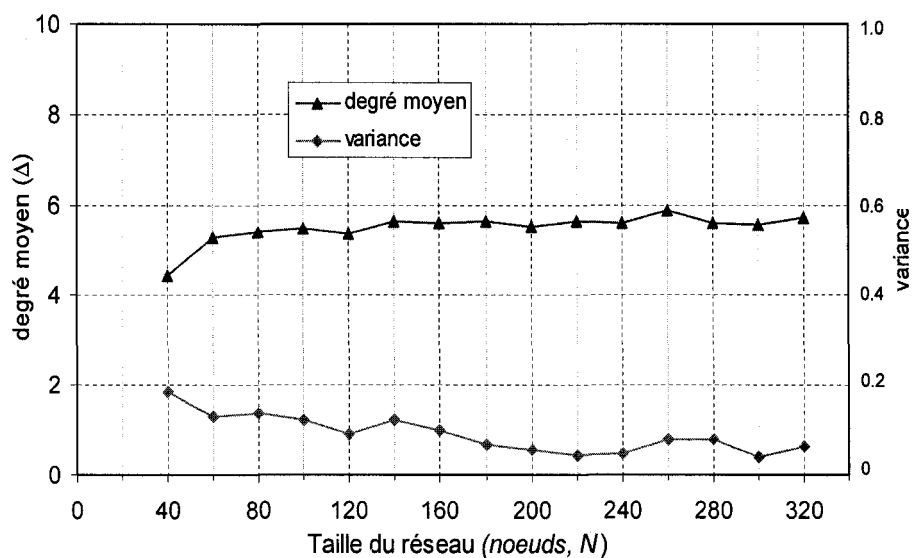


Figure 24 Degré moyen et sa variance en fonction de la taille du réseau N

La figure 25 illustre un exemple de graphe généré par la procédure décrite dans cette section pour $N = 60$. Dans cet exemple, le degré moyen pour ce graphe $\Delta = 4.8$ et le nombre d'arrêtes total $m = 143$. L'ensemble des nœuds dominants (MDS), déterminés par l'étape 1 de l'algorithme, comprend 11 nœuds $\{11, 12, 14, 15, 20, 21, 24, 25, 26, 27 \text{ et } 31\}$. Les nœuds intermédiaires, déterminés par l'étape 2 de l'algorithme, comprennent 14 nœuds $\{3, 6, 7, 23, 29, 30, 33, 34, 40, 49, 51, 54, 58 \text{ et } 60\}$. La taille de la dorsale est égale à 25. En utilisant les deux autres approches *CDS-based* et *WCDS*, on retrouve pour le même exemple une taille égale à 28 et 27 respectivement.

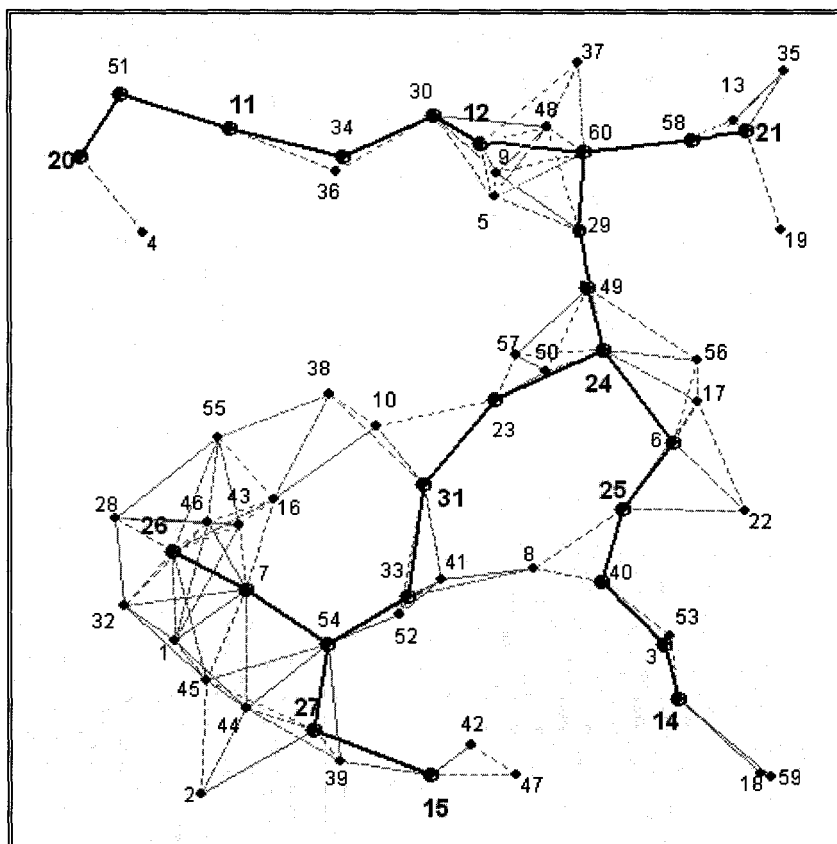


Figure 25 Graphe g n r  pour $N = 60$ et $\Delta = 4.8$

Pour comparer les trois approches (MDS-based, WCDS, CDS-based et B-CDS dans ces deux versions centralis e et distribu e) nous allons d terminer la taille de l'ensemble de domination connexe MCDS en fonction de la taille du graphe. Pour cela, nous prendrons des graphes dont la taille varie entre 40 et 320 n uds ($40 \leq N \leq 320$). Pour chaque valeur de N , nous effectuerons plusieurs simulations afin d'obtenir une moyenne statistiquement confiante de la valeur moyenne de la taille de MCDS.

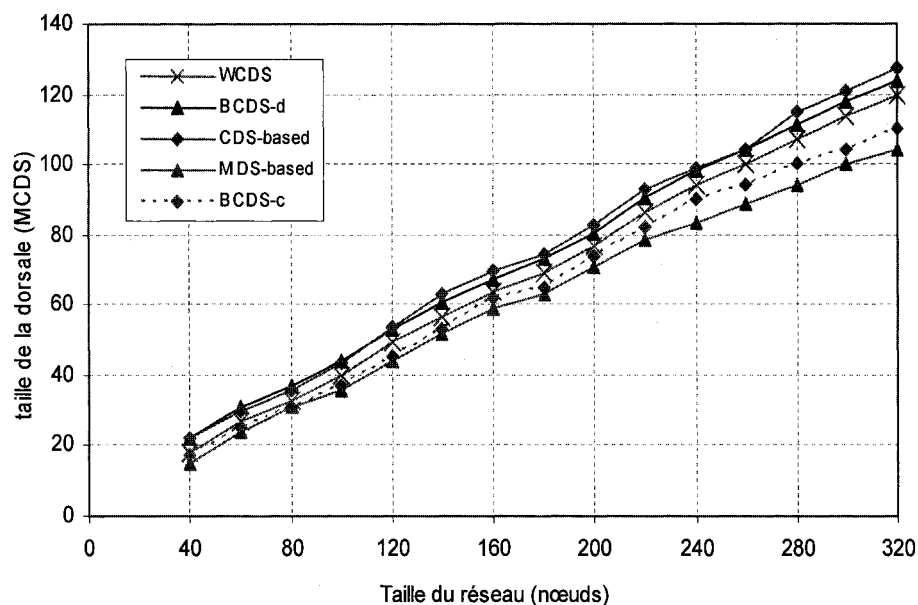


Figure 26 Taille moyenne de l'ensemble MCDS en fonction de la taille du réseau

La figure 26 illustre la taille de l'ensemble MCDS calculé en utilisant les algorithmes WCDS, CDS-based et BCDS dans ces deux versions centralisée et distribuée décrits dans la section 2.5.3, ainsi que notre algorithme présenté dans ce chapitre MDS-based, en fonction de la taille du réseau. Ces résultats de simulation montrent bien que les algorithmes centralisés (MDS-based et BCDS-c) donnent une taille de l'ensemble MCDS beaucoup plus petite comparé aux algorithmes distribués (WCDS, CDS-based et BCDS-d). Ceci est expliqué par le fait que les algorithmes distribués ne possèdent pas une information globale du réseau comme dans le cas des algorithmes centralisés. Par ailleurs, l'algorithme proposé dans ce chapitre (MDS-based) fournit une solution minimale de l'ensemble MCDS comparé à l'algorithme BCDS dans sa version centralisée. L'écart est nettement remarquable quand la taille du réseau est importante. Ces résultats prouvent que notre algorithme donne une solution minimale étant donné que la formulation de la première étape donne une solution exacte de l'ensemble de domination de taille minimale. Notre algorithme, dans sa première étape, garantit une

solution minimale de l'ensemble des dominants. En effet, il minimise la taille de l'ensemble de dominants tout en sélectionnant les nœuds ayant le plus haut degré (ayant plus de voisins) tout en dans le réseau. La figure 27 illustre le degré moyen des nœuds sélectionnés par chaque algorithme pour former l'ensemble MCDS en fonction de la taille du réseau. Ces résultats montrent bien que les nœuds sélectionnés par notre algorithme (MDS-based) possèdent le plus haut degré. Pour un même nombre de nœud dans un réseau, en choisissant les nœuds ayant plus de voisins on aura un ensemble de domination connexe de taille plus faible.

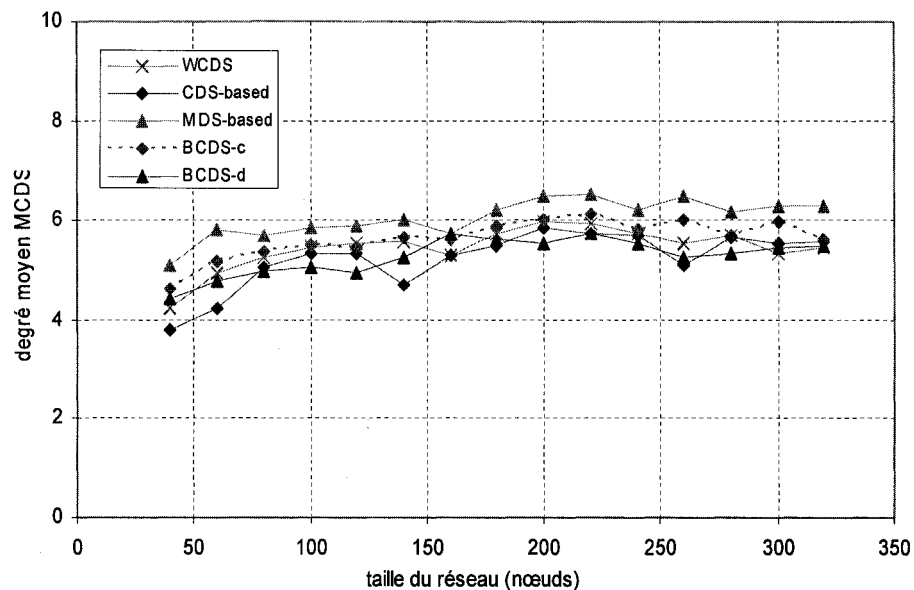


Figure 27 Degré moyen en fonction de la taille du réseau

Au niveau de la complexité, le temps d'exécution de la première étape de notre algorithme est de l'ordre de $O(n)$ et le temps d'exécution de la deuxième étape est de l'ordre de $O(m+n \cdot \log n)$. Alors la complexité totale de notre algorithme MDS-based est de l'ordre de $O(m+n \cdot \log n)$ dans le pire cas. Le tableau suivant compare les trois algorithmes en termes de temps d'exécution et de facteur d'approximation par rapport à la solution optimale. Le facteur d'approximation est défini comme étant le rapport de la taille de la dorsale construit par la taille de l'ensemble MCDS.

Tableau IV

Comparaison de la complexité des différents algorithmes

| | W-CDS | CDS-based | B-CDS | | MDS-based |
|-------------------------|----------------------|-------------------------|-----------------|------------|-------------------|
| Type | distribué | distribué | distribué | centralisé | centralisé |
| Temps d'exécution | $O(\Delta_{\max}^3)$ | $O(n^2)$ | $O(n \log^3 n)$ | $O(mn)$ | $O(m + n \log n)$ |
| Facteur d'approximation | $2 + \ln \Delta$ | $1 + \ln \Delta_{\max}$ | | | δ |

Δ_{\max} : degré maximum dans le graphe G

δ : approximation de la solution pour l'arbre de recouvrement minimum ($\delta = 2 + \varepsilon$, avec $\varepsilon > 0$) [Arora et Karakostas (2000)]

3.5 Conclusion

Dans la première partie de ce chapitre, nous nous sommes intéressés à la construction d'une dorsale virtuelle basée sur l'approximation de l'ensemble de dominations connexe de taille minimale (MCDS) dans un graphe UDG. Nous avons proposé un algorithme (MDS-based) composé de deux étapes; la première étape consiste à trouver l'ensemble des nœuds dominants (MDS) dans un graphe et la deuxième étape détermine l'arbre de recouvrement d'un graphe réduit contenant l'ensemble MDS pour, finalement, extraire l'ensemble des nœuds qui forme la dorsale.

La deuxième partie était consacrée à l'étude de performance de notre solution. En utilisant un modèle probabiliste, nous avons vérifié à l'aide des simulations que la solution fournie par notre algorithme est très proche de celle trouvée par analyse. Les simulations ont été faites en variant le nombre de nœuds dans le réseau et le rayon de transmission des nœuds.

Dans la dernière partie de ce chapitre, nous avons comparé notre algorithme avec trois autres algorithmes proposés dans la littérature. Les résultats de simulation montrent que

notre approche performe mieux que les autres algorithmes en fournissant un ensemble MCDS de taille plus petite. Comme nous l'avons mentionné auparavant, plus la taille de l'ensemble des nœuds qui forment la dorsale est minimale plus le résultat est meilleur.

Dû à l'absence d'une infrastructure fixe dans les réseaux *ad hoc*, les nœuds doivent échanger les messages de contrôle utilisés par les protocoles de routage. Le trafic de contrôle génère un trafic redondant et consomme une bonne partie de la bande passante, ressource rare dans les réseaux *ad hoc*. Une dorsale virtuelle construite à partir de l'ensemble MCDS permet de réduire l'ensemble des nœuds qui sera désigné pour retransmettre les messages de contrôle. Ainsi, le trafic de contrôle sera minimisé. Dans le chapitre suivant, nous allons évaluer notre approche comme technique de diffusion en la comparant avec d'autres techniques de diffusion utilisées dans les réseaux *ad hoc* tels que l'inondation pure, la technique de relais multipoint etc.