Université Pierre et Marie Curie

Laboratoire d'informatique de Paris VI

Routage QoS dans les réseaux ad hoc

Stéphane LOHIER

DEA Réseaux

Responsable : Pr. Guy PUJOLLE

Encadrement : Sidi-Mohammed SENOUCI
Anelise FONSECA
Nadjib ACHIR

Avril-septembre 2002

Sommaire

Résumé

L'utilisation de réseaux locaux sans fils est en nette progression depuis la commercialisation de cartes et de stations de base à la norme IEEE 802.11. Les débits atteints par ces réseaux (11 à 54 Mbit/s) permettent d'exécuter des applications multimédia nécessitant des garanties sur le débit, le délai ou encore la bande passante. Les travaux réalisés pour apporter de la qualité de service aux réseaux filaires (IntServ, RSVP ou DiffServ) ou sans fils avec point d'accès (Mobile IP ou UMTS) ne peuvent être transposés directement aux réseaux ad hoc, dans lesquels aucune infrastructure fixe n'est présente. Plusieurs approches existent pour apporter de la QoS à ces réseaux. L'adaptation des protocoles de routage classiques en est une.

Après quelques rappels sur l'architecture des réseaux ad hoc, cette étude présente les principaux protocoles de routage sans fils avec une comparaison sur leurs performances. La troisième partie est consacrée aux extensions QoS proposées à partir des protocoles existants. Une proposition de protocole QoS est ensuite présentée avec quelques éléments de simulation.

1. Généralités sur les réseaux mobiles ad hoc

1.1. Eléments d'architecture

Un réseau ad hoc est caractérisé par une architecture adaptative en fonction du nombre de stations, de leur situation géographique et de leur mobilité. Dans la mesure où les stations ne sont pas toutes directement connectées et sont susceptibles de se déplacer, il n'existe pas d'infrastructure fixe [1] de type liaisons filaires et routeurs ou liaisons sans fil et station de base. Les transmissions sont sans fil (liaisons radio), à portée limitée et sujettes à perturbations et interférences. Les liens peuvent être unidirectionnels ou bidirectionnels suivant la puissance des émetteurs et l'éloignement.

Parmi les protocoles ad hoc de niveau liaison, on retrouve Hyperlan ou 802.11 qui prévoit un mode ad hoc à la couche 2 (voir annexe) . La couche réseau est basée sur IP et un protocole de routage. Le groupe de travail MANET (*Mobile Ad hoc NETwork*) propose un certain nombre de ces protocoles en cours de normalisation.

Les applications sont diverses (fichiers, informations, multimédia, temps réel…) et peuvent être envisagées dans de nombreux domaines (armée, conférence, embarqué, médical, domicile, Internet ambiant…) [5].

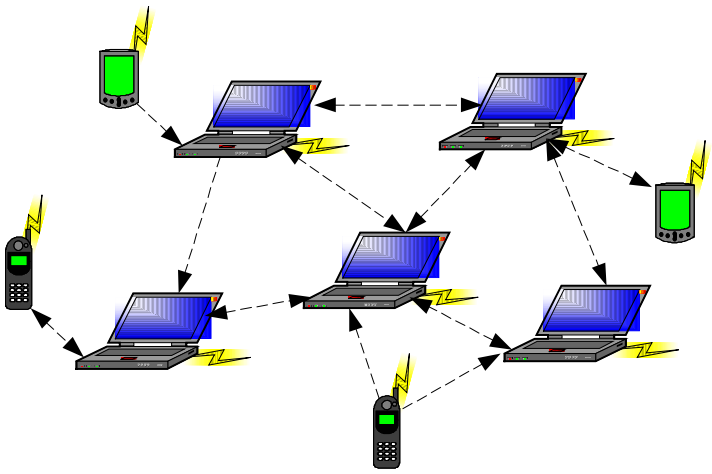


Figure 1 - Architecture de réseau ad hoc

1.2. Caractéristiques du routage

L'adressage au niveau 2 ou 3 et par suite la commutation ou le routage utilisé par les réseaux filaires est inadapté aux contraintes des réseaux ad hoc (mobilité, ressources limitées, liaisons sans fil…). Des protocoles de routage spécifiques doivent être mis en place pour établir une route de manière dynamique entre deux nœuds communiquant et maintenir cette route quelle que soit la mobilité des nœuds qui affecte de manière imprévisible la topologie du réseau.

Dans la mesure où les portées d'émission et de réception sont limitées, chaque nœud est susceptible, à un instant donné, de jouer le rôle de routeur pour relayer la communication entre deux nœuds éloignés du réseau [2] (l'utilisation de plusieurs sauts peut être aussi un facteur de réduction de la puissance d'émission nécessaire pour transmettre en un point éloigné).

Le choix de l'algorithme de routage [4] est complexe car il doit prendre en compte de nombreux paramètres :

- l'absence de centralisation (distribution des informations de routage) ;
- la taille du réseau (évolutive) ;
- la mobilité, la connectivité et la topologie ;
- le trafic utilisateur ;
- des ressources limitées (CPU, mémoire, batteries…) ;
- les contraintes des couches inférieures (liens unidirectionnels) ;
- le multicast ;
- la sécurité ;
- la qualité de service (QoS).

1.3. Qualité de service

La qualité de service dans les réseaux ad hoc peut être introduite à plusieurs niveaux interdépendants [15] :

- au niveau des protocoles d'accès au médium en ajoutant des fonctionnalités aux couches basses afin de pouvoir offrir des garanties ;
- au niveau des protocoles de routage, en recherchant des routes plus performantes suivant différents critères ;
- au niveau de la signalisation avec des mécanismes de réservation de ressources indépendants du protocole de routage ;
- dans un modèle d'architecture globale, en cherchant une cohérence avec les réseaux fixes au travers, par exemple, du contrôle par politique en intégrant des éléments de l'implémentation de ces politiques (PEP) dans les nœuds.

La QoS au niveau signalisation ou dans un modèle global est responsable de la coordination de l'implantation des autres niveaux de QoS (MAC et routage) ainsi que d'autres composantes, telles le séquencement (*scheduling*) ou le contrôle d'admission (figure 2).

Le but du routage avec qualité de service est de déterminer une route avec suffisamment de ressources disponibles pour satisfaire une requête. La réservation effective de ressources sur la route optimum évaluée par le protocole de routage est généralement laissée à la partie signalisation ou à un modèle plus général. Par ailleurs, l'implantation de QoS au niveau signalisation ne peut fonctionner sans se préoccuper du routage dans la mesure où une réservation peut échouer si les ressources ne sont pas disponibles sur une route ou un lien.

La QoS au niveau MAC est un composant essentiel du support de la QoS dans un réseau ad hoc. Les composants QoS des couches supérieures (routage et signalisation) sont dépendantes et coordonnées avec le protocole de QoS de niveau MAC [19].

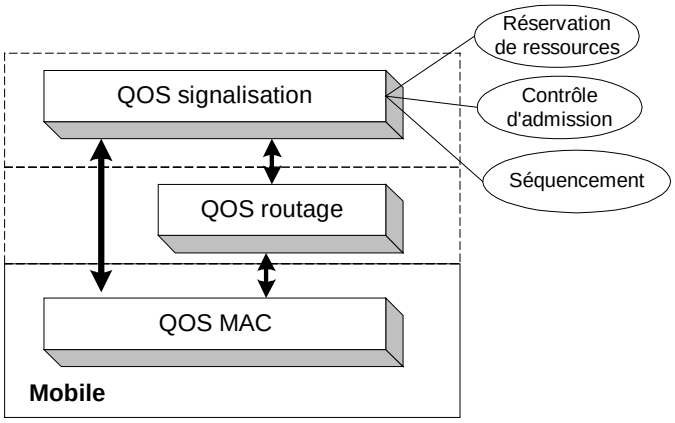


Figure 2 - Modèle QoS

Plusieurs métriques de routage QoS peuvent être utilisés : le délai, le débit, la bande passante ou encore le coût (nombre de sauts, ressources sollicitées à chaque nœud…).

Quelque soit le ou les métriques choisis, la mobilité rend difficile le respect des contraintes de QoS pendant toute une communication et il est nécessaire de tenir compte de la durée de vie des liens.

Un autre problème spécifique aux réseaux ad hoc est la limitation de la durée d'utilisation des batteries de chaque nœud. Le respect des contraintes de QoS et plus précisément de la durée de vie des routes choisies passe également par la gestion des ressources en énergie de l'ensemble.

Par ailleurs, les modèles de QoS des réseaux filaires (basés sur IntServ, RSVP ou DiffServ) reposent sur un certain nombre de constantes telles une topologie stable, des pertes faibles ou une bande passante disponible large ou extensible et sont peu adaptés aux contraintes des réseaux ad hoc (les algorithmes de calcul de métrique, bande passante ou délai, peuvent néanmoins être transposés). Les protocoles de routages spécifiques du monde ad hoc doivent donc être étendus ou modifiés pour intégrer la QoS suivant les différents métriques.

2. Principaux protocoles de routage ad hoc

2.1. Différentes approches

Les protocoles de routage peuvent être classés en trois catégories [1] :

- Les protocoles **proactifs** (*table-driven*) qui réalisent une évaluation permanente des routes dans le réseau par diffusion périodique de messages. On retrouve dans cette catégorie les algorithmes de type vecteur de distance (*distance vector*) et état des liens (*link state*) utilisés dans les réseaux fixes.
- les protocoles **réactifs** (*on-demand*) qui engagent une procédure de découverte de route sur demande seulement. Les algorithmes de recherche par inondation (*flooding*) sont utilisés.
- les protocoles **hybrides** qui combine les caractéristiques des deux types. Les nœuds maintiennent de manière proactive une information de topologie locale et le routage est réalisé selon une technique réactive.

D'autres distinctions peuvent être réalisées à l'intérieur de ces catégories [5]. On distingue ainsi les protocoles qui introduisent une hiérarchie entre les nœuds (*hierarchical protocols*) de ceux qui attribuent le même rôle à tous les nœuds (*flat protocols*). Certains protocoles ont en outre recours à des informations sur la localisation géographique des nœuds (*Physical Location Information -PLI-based protocols*).

Les principaux protocoles sont détaillés dans la suite.

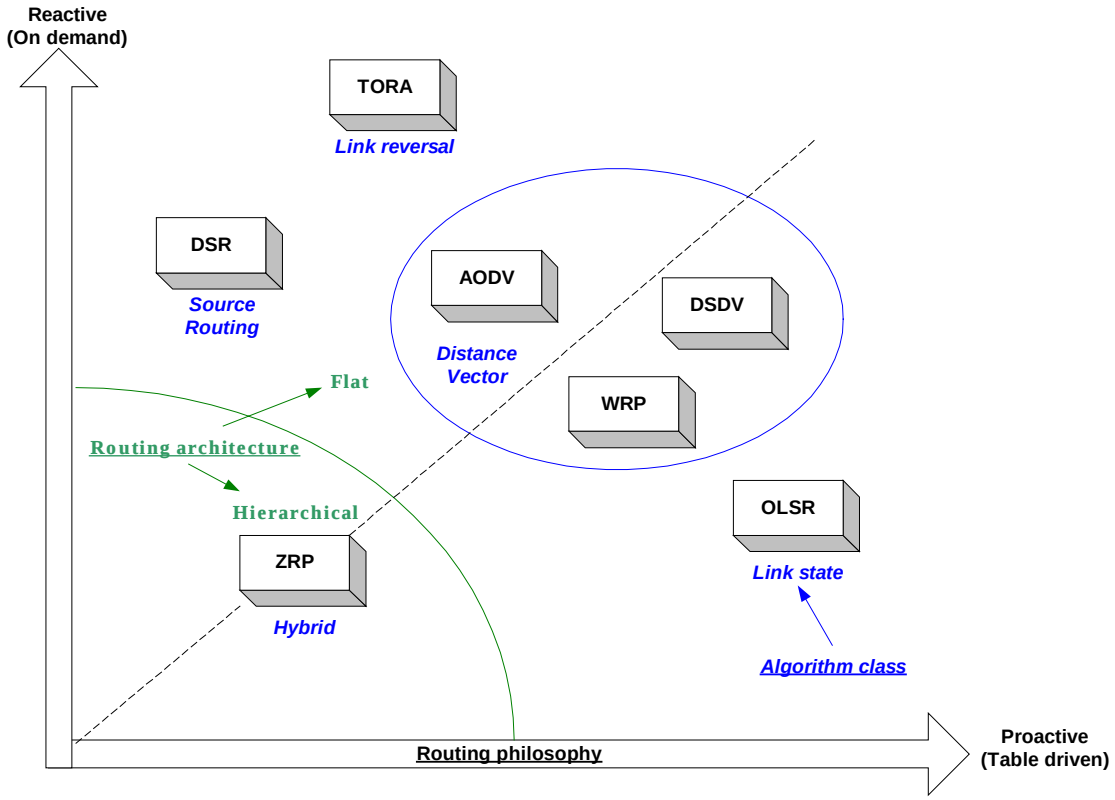


Figure 3 - Classification des protocoles de routage ad hoc

2.2. DSDV (*Destination Sequenced distance Vector*) –1994

DSDV [11] est un protocole proactif de première génération basé sur l'idée classique de l'algorithme distribué de Bellman-Ford (DBF) en ajoutant une gestion séquentielle des paquets de mise à jour. Il sert de base à d'autres protocoles plus évolués tels que AODV.

Chaque station mobile maintient une table de routage qui contient pour chaque entrée :

- l'adresse destination ;
- le prochain nœud (*Next hop*);
- le nombre de sauts (*Metric*) ;
- le numéro de séquence attribué par la destination (*Dest_SN*).

Le *Dest_SN* permet de distinguer la plus récente mise à jour afin d'éviter la formation des boucles de routage (*routing loop*) et le comptage infini (*counting to infinity*).

Un paquet de mise à jour est transmis périodiquement par un nœud à tous ses voisins. Il contient :

- le numéro de séquence du nœud émetteur (*Src_SN*) qui permet de référencer de manière unique chaque diffusion.

Et pour chaque route :

- l'adresse destination ;
- le nombre de sauts (*Metric*) ;
- le numéro de séquence attribué par le nœud destination (*Dest_SN*).

La table de routage est mise à jour si :

- la nouvelle route entrante possède un *Dest_SN* plus récent ;
- le *Dest_SN* entrant est identique mais le nombre de sauts est inférieur (cas de l'exemple).

Deux méthodes de mise à jour sont possibles :

- la totalité de la table de routage du nœud est transmise périodiquement aux voisins *(full dump) ;*
- seules les entrées modifiées de la table de routage du nœud sont diffusées (*incremental*) ; cette méthode est préférée lorsque le réseau est relativement stable.

De plus, pour limiter le trafic, une station attend un temps basé sur le délai moyen entre deux mises à jour avant de diffuser une nouvelle route choisie.

La figure 4 montre partiellement une mise à jour de la table de routage d'un nœud M4 après déplacement d'un nœud M1 ; la nouvelle route choisie est celle de plus petit métrique. Les numéros de séquence transmis par les autres stations peuvent avoir évolué entre-temps (cas de M3).

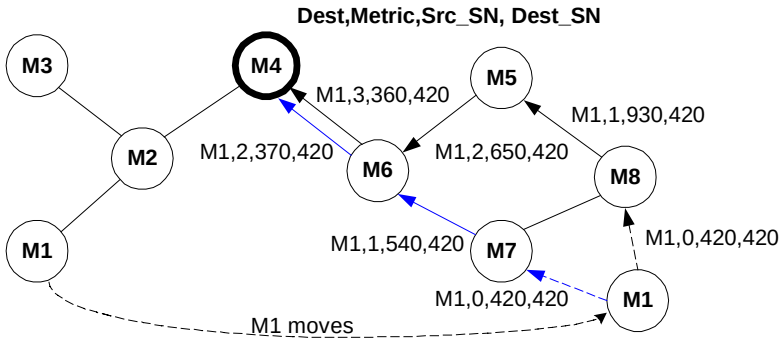


Table de routage de M4 :

Avant déplacement de M1

Destination	NextHop	Metric	Dest_SN
M1	M2	2	410
M2	M2	1	780
M3	M2	2	250

Après déplacement de M1

Destination	NextHop	Metric	Dest_SN
M1	M6	3	420
M2	M2	1	780
M3	M2	2	260

Figure 4 - Exemple d'échange DSDV

Performances

DSDV est caractérisé par sa lenteur de convergence des routes et une surcharge permanente de trafic due à des mises à jour périodiques, ce qui est d'autant plus pénalisant pour une faible mobilité.

2.3. DSR (*Dynamic Source Routing*)

DSR [8, 10] est un protocole réactif basé sur le concept du routage à la source (*source routing)*. Deux procédures successives sont mises en place : découverte de routes et maintenance.

Découverte de routes (*route discovery*)

Un nœud émetteur regarde d'abord dans son cache si une route vers la destination est présente.

- Si une route existe le nœud diffuse le paquet en utilisant l'algorithme de routage à la source (*source routing).*
- Si aucune route n'existe, le nœud diffuse un paquet de découverte RREQ (*Route REQuest)* qui contient :

- l'adresse source (*Src*) ;
- l'adresse destination (*Dest*) ;
- la liste des nœuds traversés pour atteindre la destination (*List*), vide au départ ;
- un identifiant unique pour qu'un nœud ne propage que les paquets nouveaux.

Sur réception d'un RREQ(*Src,Dest,List*), un nœud N mémorise la liste dans son cache pour l'enrichir et vérifie si son adresse correspond à la destination.

- Si ce n'est pas le cas et s'il n'a pas déjà reçu le paquet, il consulte son cache :
  - si la route vers le nœud destination est présente, il transmet un paquet de réponse RREP (*Route REPly*) vers la source ;
  - sinon, il fait suivre le paquet de découverte à ses voisins en rajoutant sa propre adresse à la liste : RREQ(*Src,Dest,List+N*).
- Si le nœud est la destination, il diffuse un paquet de réponse RREP.

Pour remonter le RREP, le nœud destination regarde dans son cache si une route vers la source est présente.

- Si une route existe, le nœud diffuse le paquet en utilisant à son tour l'algorithme de routage à la source (*source routing).*
- Si aucune route n'existe, deux cas peuvent se présenter :
  - les liens sont bidirectionnels, la destination transmet le RREP en inversant la liste des nœuds donnée dans le RREQ ;
  - les liens ne sont pas symétriques, la destination diffuse une recherche de route avec un nouveau RREQ en encapsulant (*piggyback*) le RREP : RREQ(*Src,Dest*,RREP()).

Les nœuds intermédiaires retransmettent le RREP en mémorisant également la liste dans leurs caches.

La figure 5 montre une découverte de route suivie par une réponse de la destination. Les nœuds recevant les RREQ de plusieurs voisins ne font suivre que le premier RREQ pour éviter les redondances.

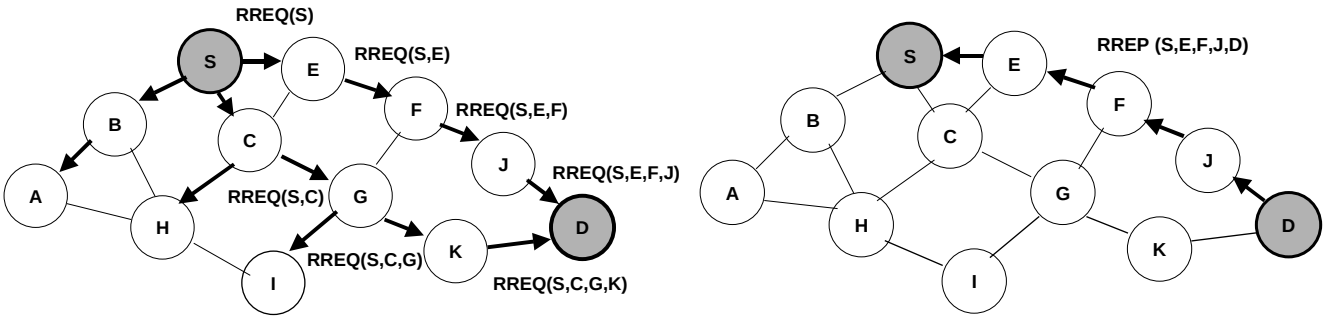


Figure 5 - Exemple de découverte DSR

Maintenance de route (*route maintenance*)

La maintenance de route est réalisée en utilisant des paquets d'erreur et des acquittements.

Lorsqu'un nœud se trouvant sur la route empruntée par les données ne répond pas (figure 6), après plusieurs tentatives de retransmission, un paquet d'erreur RERR contenant les adresses des deux extrémités du lien défaillant est généré vers la source.

Les nœuds en amont qui reçoivent le RERR suppriment le nœud en erreur de leur cache et toutes les routes contenant le lien défaillant sont modifiées en conséquence.

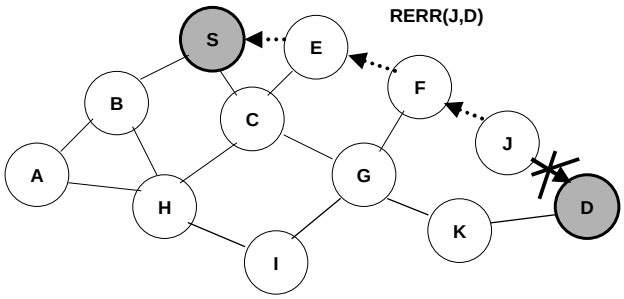


Figure 6 - Exemple de rupture de lien DSR

L'absence de réponse d'un nœud peut être est détectée par la couche MAC si une procédure d'acquittement de proche en proche est prévu à ce niveau ou par l'utilisation d'un acquittement passif si la station est capable d'entendre la suite de la transmission du paquet entre les deux nœuds suivant. Une dernière possibilité est l'utilisation d'acquittements spécifiques à la couche réseau et gérés par DSR de bout en bout.

Performances

Le routage à la source permet de repérer immédiatement les boucles et de les éliminer. De plus, les nœuds apprennent dynamiquement les routes en scrutant les paquet RREQ, RREP ou de données qu'ils reçoivent ce qui évite une transmission périodique des mises à jour.

Par contre, DSR introduit une surcharge croissante dans chaque paquet de données qui contient le chemin complet vers la destination ainsi qu'un trafic inutile lié à la diffusion des paquets RREQ (un nœud donné reçoit inutilement un même RREQ des tous ses voisins).

Les liens asymétriques constituent un autre inconvénient majeur de ce protocole car dans ce cas, une nouvelle procédure de découverte doit être initiée par la destination.

Par ailleurs, les caches n'intègrent pas de métriques pour minimiser la longueur des routes.

2.4. AODV (Ad hoc On demand Distance Vector)

AODV [6, 12] est un protocole de routage réactif, qui utilise un mécanisme de découverte de route inspiré de DSR et un algorithme de routage similaire à celui de DSDV. Une fois la route tracée, les nœuds qui ne se trouvent pas sur le chemin actif ne maintiennent aucune information de routage et ne participent à aucun échange de mise à jour.

Chaque nœud maintient une table de routage ayant une entrée par route active qui contient :

- l'adresse destination (*Dest*) ;
- le prochain nœud (*Next hop*);
- le nombre de sauts (*Hop Count*) ;
- le numéro de séquence attribué par la destination (*Dest_SN*) ;
- la liste des prédécesseurs (nœuds voisins auxquels une réponse RREP à été transmise et qui pourront être alertées en cas de rupture de lien en aval) ;
- un temporisateur d'expiration pour la route (réarmé dès qu'un paquet est reçu sur la route, son expiration rend la route obsolète et l'efface de la table) ;

Recherche d'une route (*path discovery*)

Un nœud qui n'a pas de chemin valide dans sa table pour une destination voulue diffuse à ses voisins un paquet RREQ qui contient :

- l'adresse de la source (*Src*) et de la destination (*Dest*) ;
- le numéro de séquence de la source (*Src_SN*) et de la destination (*Dest_SN*) ;
- le nombre de sauts (*Hop Count*) ;
- l'identifiant de la diffusion (*broadcast ID)* qui est incrémenté à chaque retransmission du RREQ.

Les nœuds suivants ignorent un RREQ déjà reçu (*Src* et *broadcast ID* identifient de manière unique une requête).

Sinon, chaque nœud atteint par un RREQ cherche une route dans sa table de routage pour la destination.

- S'il ne possède pas de route active assez récente (le *Dest_SN* reçu dans le RREQ est supérieur au *Dest_SN* mémorisé dans la table) le nœud diffuse le RREQ à son tour en incrémentant le nombre de sauts (en l'absence de réponse RREP au bout d'un certain temps, le nœud diffuse de nouveau le RREQ en incrémentant le *broadcast ID*).
- Sinon, il met à jour sa table de routage selon le RREQ reçu et renvoie un paquet de réponse RREP (*Route REPly)* vers la source. Le RREP contient :
  - l'adresse de la source (*Src*) et de la destination (*Dest*) ;
  - le numéro de séquence de la destination (*Dest_SN*) ;
  - le nombre de sauts (*Hop Count*).

Les nœuds recevant en retour le RREP mettent à jour leur table et font suivre le paquet vers la source, qui commence à émettre ses données lorsqu'elle le reçoit le premier RREP. La source changera de route si un RREP reçu par la suite lui en apprend une meilleure (*Dest_SN* supérieur ou nombre de sauts inférieur). Pour maintenir la connaissance du chemin inverse et faire suivre un éventuel paquet RREP, les nœuds traversés lors de l'envoi du RREQ gardent en mémoire (pendant un temps basé sur le *Src_SN*) l'adresse du voisin ayant transmis la première copie du RREQ. De même, pour mémoriser le chemin direct, les nœuds traversés par un RREP maintiennent un pointeur vers le nœud d'où vient le RREP.

L'exemple suivant montre une découverte de route suivie par une réponse de la destination lorsque qu'aucun nœud intermédiaire n'a trouvé de route valide (dans ce cas , *Dest_SN* est mis à jour par D).

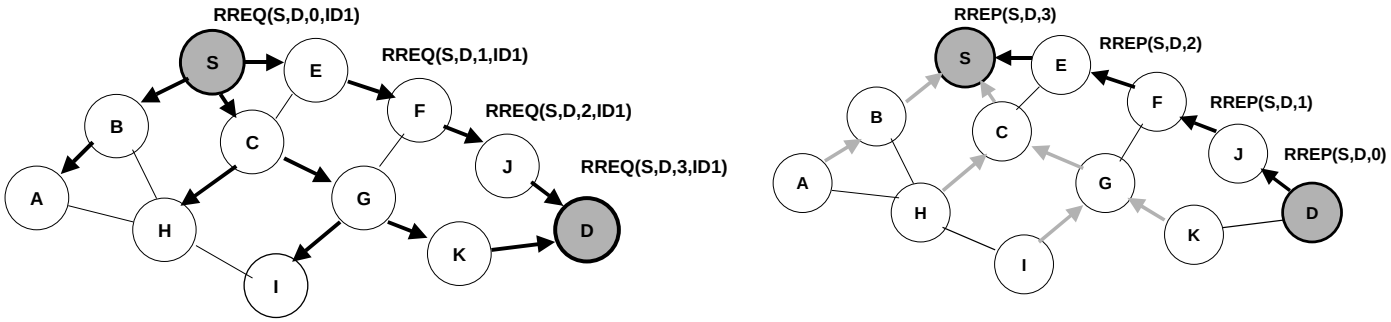


Figure 7 - Exemple de recherche de route AODV

Maintenance des routes (*path maintenance*)

Dans le cas du déplacement d'un nœud source, il suffit que celui-ci relance une procédure de découverte pour établir une nouvelle route.

La détection des ruptures de liens peut être réalisée à l'aide de messages spécifiques « *hello* » diffusés périodiquement d'un nœud vers ses voisins immédiats ou par l'écoute de la transmission d'un paquet de données sur le lien suivant.

Le nœud qui détecte une rupture de lien pour le nœud suivant d'une route active (ou qui reçoit un paquet de données à destination d'un nœud pour lequel il ne possède pas de route active) diffuse un message RERR (*Route ERRor*).

Les paquets RRER contiennent les informations suivantes :

- L'adresse (les adresses) de la (des) destination(s) inaccessible(s) (*Unreachable Dest*) ;

- le numéro de séquence de la destination inaccessible (*Unreachable Dest_SN*)
- le nombre de destinations inaccessibles (*DestCount*) ;

Les nœuds recevant un paquet RERR le diffuse à leur tour s'il provient d'un nœud en aval sur la route active.

Dans certains cas, si la rupture à lieu sur un lien pas trop éloigné de la destination (nombre de sauts inférieur à une valeur prédéfinie), le nœud situé en amont de la rupture du lien et qui a détecté celle-ci va d'abord tenter de réparer localement le lien en relançant un RREQ vers la destination. Si la réparation locale est effective (au moins un RREP est reçu en retour), le nœud averti les stations en amont avec un message RRER spécifique (bit N de l'en-tête à 1) pour que ces derniers n'effacent pas la route. Sinon, le message RERR standard est diffusé comme décrit précédemment.

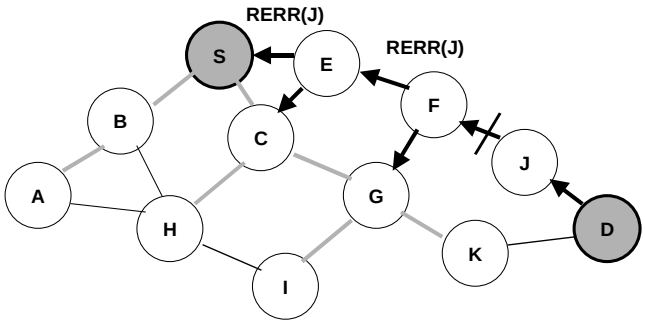


Figure 8 - Exemple de rupture de lien AODV

Performances

Comme pour DSR, les nœuds apprennent dynamiquement les routes à l'aide des paquets de découverte ce qui évite les messages systématiques de mise à jour des tables utilisés dans DSVD. AODV présente aussi l'avantage d'éliminer la surcharge due au routage par la source de DSR. Un fonctionnement multicast est également prévu.

Parmi les principaux inconvénients d'AODV :

- les liens doivent impérativement être symétriques ;
- le choix de la route la plus récente ne conduit pas forcément à la route optimale ;
- une seule route est gardée par destination, ce qui crée une latence lors de ruptures de liens ;
- une surcharge importante est générée par le recours fréquent à des phases de découvertes de routes, par la diffusion d'un même RREQ à un nœud donné et par l'utilisation éventuelle de messages *hello*.

2.5. ZRP (*Zone Routing Protocol*)

ZRP [13] est un protocole de routage hybride et hiérarchique qui combine à la fois :

- une approche proactive à l'intérieur d'une zone restreinte, ce qui permet de mettre à jour l'état du réseau et de maintenir des routes qu'il y ait ou non des paquets de données qui circulent.
- une approche réactive entre les zones restreintes, qui ne détermine une route entre des nœuds périphériques que si le besoin de transmettre des paquets de données apparaît.

Routage intra-zone.

L'étendue d'une zone est définie par rapport à un nœud central. Tous les nœuds ayant une distance d'au plus r (rayon de zone - *zone radius*) par rapport à un nœud X est considéré dans la zone de routage de X.

Le routage à l'intérieur d'une zone est réalisé par un protocole proactif IARP (*IntrAzone Routing Protocol*) basé sur un algorithme à état de lien ou à vecteurs de distance. Les nœuds ayant une distance de r sont des nœuds périphériques de la zone.

Routage inter-zone.

ZRP utilise dans ce cas un protocole réactif IERP (*IntErzone Routing Protocol)* pour trouver des chemins entre les zones. Les paquets de découverte de route sont propagés sur le même principe que pour DSR, mais les échanges entre zones se font uniquement au travers des nœuds périphériques.

La figure suivante donne un exemple de réseau utilisant ZRP. Le nœud S qui veut émettre teste la présence de la destination D. Celle-ci n'est pas dans sa zone, S diffuse alors un paquet de requête RREQ vers ses nœuds périphériques B, C et F. Ce paquet est simplement relayé par les nœuds intermédiaires A et E grâce à un protocole spécifique de « *bordercast* ». Les nœuds B, C et F testent à leur tour la présence de D dans leur voisinage. Dans la négative, ils diffusent par « *bordercast* » vers leurs nœuds périphériques. Le nœud J périphérique de F, trouve D dans sa zone de routage et répond par un paquet RREP en indiquant le chemin à suivre : S-F-J-D.

Dans cet exemple, la route est spécifiée par la liste des nœuds accumulée lors de la requête et inversée lors de la réponse comme pour le protocole DSR.

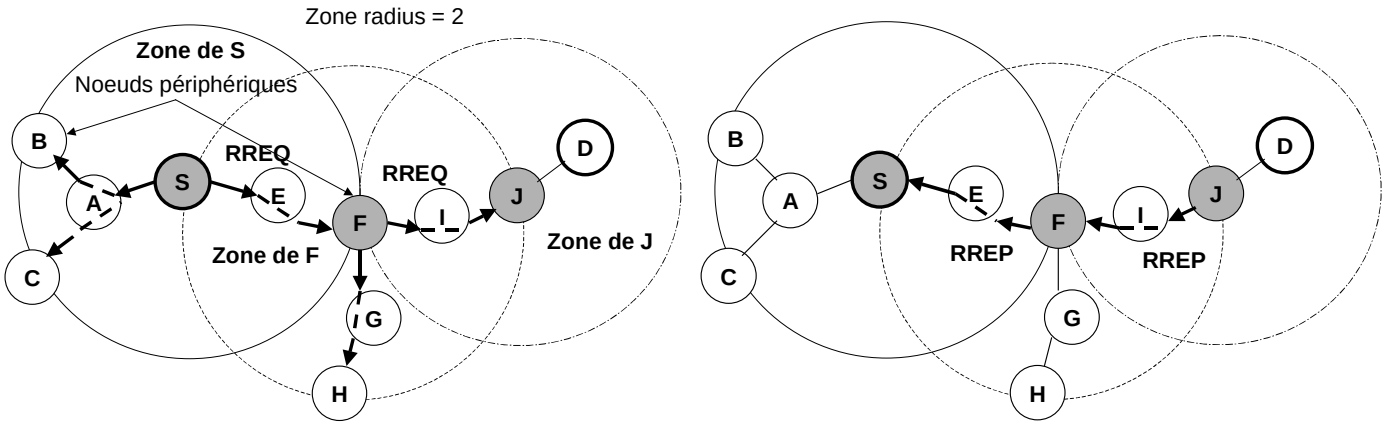


Figure 9 - Exemple de routage ZRP

Performances

L'organisation hybride et hiérarchique de ZRP permet d'exploiter les avantages des protocoles proactifs et réactifs. De plus, le rayon de zone peut être adapté en fonction de l'évolution ou de la mobilité.

Un autre avantage de ZRP est la possibilité d'obtenir de multiples réponses de route à partir d'un seul paquet de requête. La meilleure route en termes de nombres de sauts ou selon une autre métrique est ensuite sélectionnée par la source, ce qui augmente la fiabilité et les performances.

Les performances sont par ailleurs liées au protocole proactif qui n'est pas spécifié. Les changements de topologie dans une zone ne sont diffusés qu'aux voisins, alors qu'ils peuvent affecter plusieurs zones de routage. Une surcharge de trafic et des problèmes de diffusion peuvent enfin être occasionnés par le chevauchement des zones de routage (des mécanismes de contrôle sont développés pour réduire le trafic).

2.6. WRP (*Wireless Routing Protocol*) – 1996

WRP [14] est un protocole proactif de type vecteur de distance basé sur la vérification de la réponse de tous les voisins à chaque détection d'un changement dans les liens adjacents.

Chaque nœud maintient 4 tables :

- la table des distances ;
- la table de routage ;
- la table du coût des liens ;
- la table de retransmission des messages MRL (*Message Retransmission List*).

La table des distances du nœud i est une matrice qui contient pour chaque destination j et pour chaque voisin de i (k) la distance à j en passant par k et le prédécesseur de k. Cette table permet de connaître le chemin le plus court et est utilisée pour établir la table de routage.

La table de routage a une entrée par destination j et contient la distance à j, le prédécesseur et le successeur sur le chemin le plus court (*shortest path)* et un indicateur (*tag*) pour l'état du chemin (correct, boucle, pas encore déterminé).

La table du coût des liens contient pour chaque nœud i un coût vers chaque voisin k (fonction du nombre de saut et du temps de latence sur le lien) ainsi qu'une valeur de délai écoulé depuis la dernière bonne réception d'un message du voisin.

La table de retransmission des messages sert à gérer les messages de mise à jour émis par le nœud.

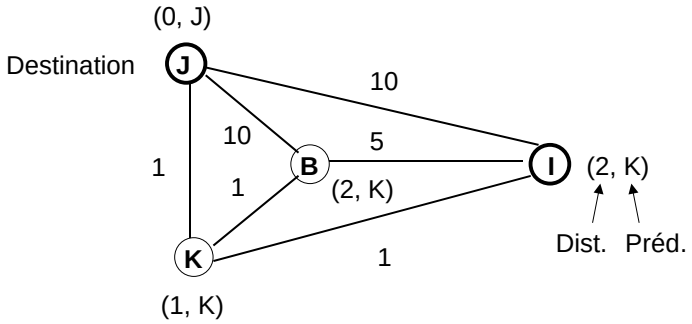
Les messages de mise à jour UPD (*UPDates*) sont échangés sur événement (changement du coût d'un lien ou rupture de lien) entre voisins. Un UPD contient la description des mises à jour à effectuer (destination, distance, prédécesseur) et /ou les acquittements à des UPD reçus des voisins, ainsi que la liste des voisins devant acquitter cet UPD. Un paquet de mise à jour non acquitté est retransmis.

Un nœud recevant un UPD d'un voisin k met à jour les champs distance et prédécesseur de l'entrée k de sa table des distances. Puis il recalcule pour toutes les destinations le chemin le plus court.

Un nœud qui n'a rien à transmettre doit émettre des messages *hello* pour assurer la connectivité. Une absence de messages *hello* en provenance d'un voisin pendant une période donnée est considérée comme une perte de connectivité. Une rupture de lien est signalée par une distance infinie dans la table des distances

Dans l'exemple suivant, les entrées des tables de distance et de routage du nœud I concernant la destination J sont données.

La table de routage est donc déterminée suivant le chemin le plus court.



Dest. J	Voisin	Dist.	Préd.
	B	12	K
	J	10	I
	K	2	K

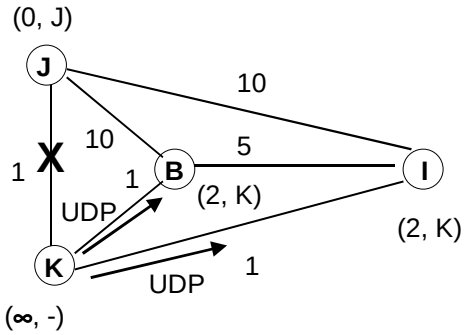
Table de distance de I

Dest.	Dist.	Préd.	Succ.	Tag
J	2	K	K	Correct

Table de routage de I

Figure 10 - Exemple de tables WRP

En cas de rupture du lien (K,J), K génère un UPD (J, ∞, -) vers B et I qui mettent à jour leurs tables de distance et de routage.



Dest. J	Voisin	Dist.	Préd.
	B	15	B
	J	10	I
	K	∞	

Table de distance de I

Dest.	Dist.	Préd.	Succ.	Tag
J	10	I	J	Correct

Table de routage de I

Figure 11 - Mise à jour des tables WRP

Les nœuds B et I émettent des UPD (contenant l'ACK pour K). L'UPD de B ne modifie pas les tables de I puisqu'il a trouvé le chemin optimal.

A partir de ces UPD, K calcule son nouveau chemin vers J et émet un UPD vers ses voisins. Cet UPD ne modifie pas les tables de routage de I et B puisqu'ils ont déjà les chemins les plus courts.

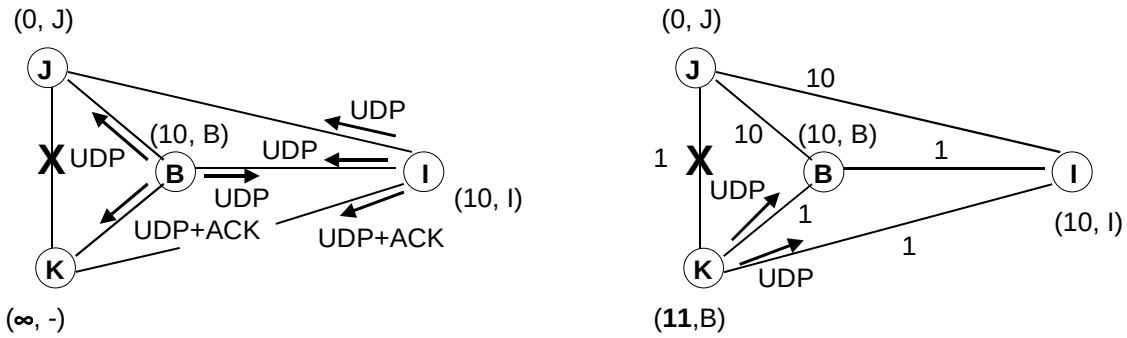


Figure 12 - Calcul des chemins WRP

Performances

WRP évite les problèmes de boucle et assure une convergence rapide des routes suite à une rupture de liens, grâce à l'utilisation de la table de distance.

Un trafic supplémentaire est occasionné par la diffusion des messages UDP et l'utilisation des messages *hello.* De plus, le maintient de 4 tables dans chaque nœud exige des ressources processeur et mémoire relativement importantes.

2.7. OLSR (*Optimized Link State Routing protocol*)

OLSR [9] est un protocole proactif basé sur un algorithme de type « état des liens ».

Pour éviter l'inondation classique (*flooding*) sur ce type d'algorithme, le protocole prévoit l'élection de nœuds spécifiques, les MPR (M*ultiPoint Relay),*chargés de transmettre les informations de topologie.

La sélection des MPR se fait à partir de messages *hello* que les nœud s'échangent mutuellement pour déduire la nature des liens, symétriques ou asymétriques, qui les relient. La condition pour devenir MPR d'un nœud est de pouvoir atteindre avec un lien symétrique, tous les nœuds voisins situés à une distance de deux sauts du nœud initial. L'ensemble des MPR d'un nœud (*MPR set*) doit donc couvrir tout le voisinage situé à deux sauts. Pour que le protocole soit efficace, le *MPR set* doit être minimisé. Après élection, les MPR sont communiqués à tout le réseau par des messages TC (*Topology Control*) périodiques. A réception des TC, les nœuds mettent à jour leur table de routage.

La limitation des diffusions est basée sur les deux règles :

- les nœuds non MPR reçoivent et exploitent les messages de diffusion mais ne les diffusent pas (sauf à leurs MPR) ;
- les MPR ne diffusent que les messages provenant des nœuds dont ils sont MPR.

Dans l'exemple suivant, les nœuds C et E sont MPR de A (ils peuvent atteindre G et H situés à 2 sauts de A) et diffusent l'information de routage reçue de A. Les nœuds E et K, MPR de H, diffusent à leur tour l'information.

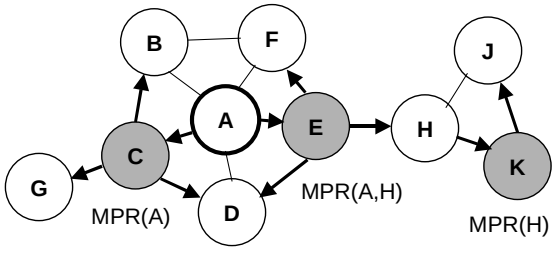


Figure 13 - Exemple d'échange OLSR

Performances

OLSR présente l'avantage de réduire la taille des messages de contrôle, car seuls les liens aux MPR sont déclarés. e plus, l'inondation est limitée par l'utilisation des MPR.

Comme pour tout protocole à état de lien, les routes utilisées sont optimales et l'absence de boucles est garantie. OLSR est adapté aux réseaux étendus, denses, et au trafic sporadique.

2.8. TORA (*Temporally Ordered Routing Algorithm*)

TORA [7] est un protocole de routage réactif, basé sur le principe de l'inversion de lien (*link reversal*). Il est prévu pour fonctionner dans un environnement de mobiles avec une forte dynamique. La recherche est initiée à la source et fournie des routes multiples pour chaque couple source/destination requis. Une fois la destination trouvée, le chemin inverse est construit.

Pour minimiser les surcharges dues aux changements de topologie, les messages de contrôle sont limités à l'ensemble des nœuds proches du lieu de changement. Les nœuds ne conservent d'information de routage que sur leurs voisins immédiats.

Trois fonctions de base sont appliquées dans TORA :

- la création de route ;
- la maintenance de route ;
- l'effacement de route.

Durant les phases de création et maintenance, les nœuds utilisent un métrique de hauteur (*height metric*), de manière à former un graphe orienté (DAG - *Directed Acyclic Graph*) dont la racine est le nœud de destination (hauteur 0).

Lors de la création de route, un nœud ne possédant pas de voisin de plus faible hauteur diffuse un paquet QRY (Query) contenant l'adresse de la destination. Le paquet se propage jusqu'à ce qu'il atteigne un nœud ayant une route pour la destination. Ce dernier répond avec un paquet UPD (*Update*) contenant sa hauteur par rapport à la destination. Un nœud recevant un paquet UPD fixe sa hauteur à une valeur plus élevée que celle du voisin qui lui a transmis le paquet. Des liens orientés (*upstream* ou *downstream*) sont ainsi créés depuis la source jusqu'à la destination en fonction des hauteurs relatives des nœuds voisins (voir figure 14).

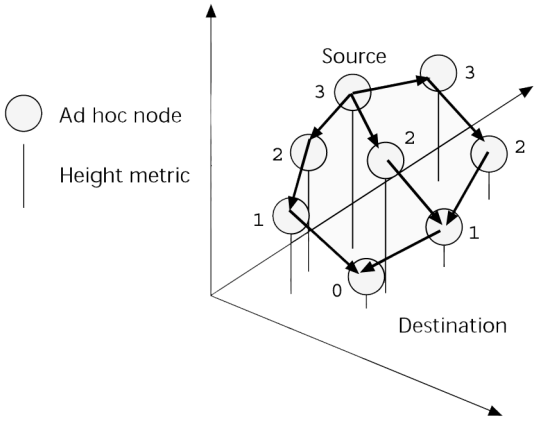


Figure 14 - Création de routes

Lorsque suite à un déplacement de nœud, une route est rompue sur le graphe, une maintenance est nécessaire pour rétablir le graphe orienté vers la même destination. Lorsqu'un nœud perd sont dernier lien descendant, il génère un nouveau niveau de référence à l'aide de paquets UDP pour indiquer à la source l'invalidité des chemins rompus. Les liens sont alors inversés (*link reversal*) pour refléter ce changement de référence (voir exemple figure 15).

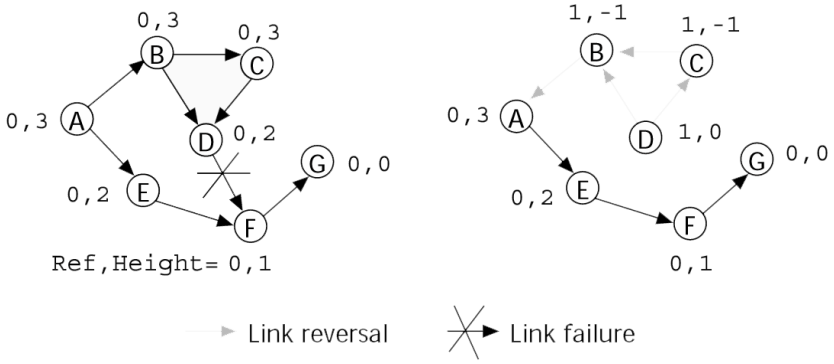


Figure 15 - Maintenance de route

Durant la phase d'effacement de routes, une diffusion de paquets CLR (C*lear*) est déclenchée pour supprimer les routes invalides.

La synchronisation est un facteur important pou TORA car le métrique de hauteur est dépendant de l'instant relatif de la rupture de lien. Ceci suppose que tous les nœuds sont synchronisé sur la même horloge (fournie par une source externe basée sur GPS par exemple).

Pour chaque nœud, cinq éléments sont définis :

- l'instant logique de la rupture de lien ;
- l'identifiant unique du nœud qui définit le nouveau niveau de référence ;
- un bit d'indication de réflexion qui permet d'indiquer un nouveau niveau ;
- un paramètre d'ordonnancement de propagation qui correspond à la hauteur du nœud ;
- l'identifiant unique du nœud.

Les trois premiers éléments représentent le niveau de référence et les deux derniers l'écart par rapport à ce niveau. Un nouveau niveau de référence est défini chaque fois qu'un nœud perd son dernier lien descendant suite à une rupture.

Performances

TORA limite la surcharge due aux paquets de contrôle en localisant leur diffusion et fournit plusieurs routes vers une destination. Des phénomènes d'oscillation peuvent cependant apparaître lorsque plusieurs ensembles de nœuds cherchent simultanément à construire de nouvelles routes ou à en effacer d'anciennes.

2.9. Comparaison des performances

Les tableaux 1 et 2 présentent une comparaison des différentes approches [5] ainsi que les caractéristiques comparées des protocoles étudiés [1].

	Avantages	Inconvénients
Proactif	Rapidité d'établissement des routes Maintien d'informations sur la qualité des routes	Routes pas toujours exploitées Convergence lente Risques de boucles Grande consommation des ressources Informations de routage sous-utilisées
Réactif	Réduction de la surcharge de routage	Délai de recherche avant émission Trafic de messages de contrôle important Possibilités de routes invalides en cache Pas d'information sur la qualité des routes
Hybrides		Latence

Tableau 1 - Comparaison des types de protocole ad hoc

Type		**Proactif**		**Hybride**		**Réactif**	
Paramètres	**DSDV**	**OLSR**	**WRP**	**ZRP**	**DSR**	**AODV**	**TORA**
Sans boucles	oui						
Nœuds critiques	non	oui	non	oui	non	non	non
Architecture du routage	flat	flat	flat	Hiérarchique	flat	flat	flat
Émissions périodiques	oui	oui	oui	oui	non	non	non
Support des liens asymétriques	non	oui	non	non	oui	non	non
Routes multiples	non	non	oui	non	oui	non	oui
Numéros de séquence	oui	non	oui	non	non	oui	non
Métrique de routage	Sauts	Sauts	Sauts	Sauts	non	Date/sauts	oui
Multicast	non	non	non	non	non	oui	non
Économie de puissance	non						
Sécurité	non						
QoS	non						

Tableau 2 - Comparaison des protocoles ad hoc

La figure 16 présente les résultats d'une simulation [3] pour quatre des protocoles décrits et permet de comparer les taux de paquets délivrés ainsi que la surcharge de paquets en fonction de la mobilité (les mesures sont effectuées pour 20 sources CBR de 2 à 32 kbit/s ; les temps de pause correspondent aux durée d'immobilisation des nœuds entre deux déplacements aléatoires à 2 m/s dans un rectangle de 1500 x 300 m, un temps de pause nul représente donc un mouvement constant). On constate que les pourcentages de paquets délivrés restent importants pour une faible mobilité ; une baisse importante apparaît pour DSDV lorsque la fréquence des déplacements augmente. Pour les protocoles proactifs, la surcharge augmente avec la mobilité et DSR obtient les meilleures performances.

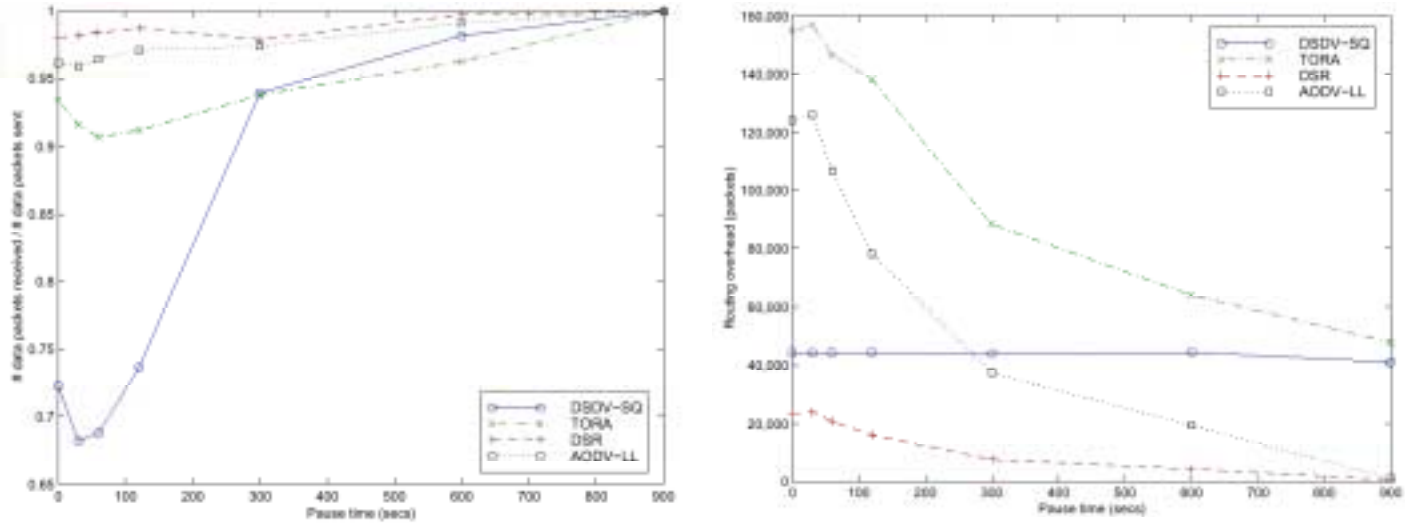


Figure 16 - Paquets délivrés et surcharge / mobilité

Les résultats de simulation [4] présentés figure 17 montrent les variations de délai et de débit en fonction de la mobilité (les mesures sont données pour 15 sources CBR à 5 paquets de 64 octets par seconde sur des liens à 2 Mbit/s ; les portées sont de 250 m et les déplacements se font toutes les secondes sur un espace de 1000 m x 1000m). Le délai de bout en bout augmente davantage avec la mobilité pour AODV. Les débits moyens obtenus par DSDV diminuent notablement avec la mobilité.

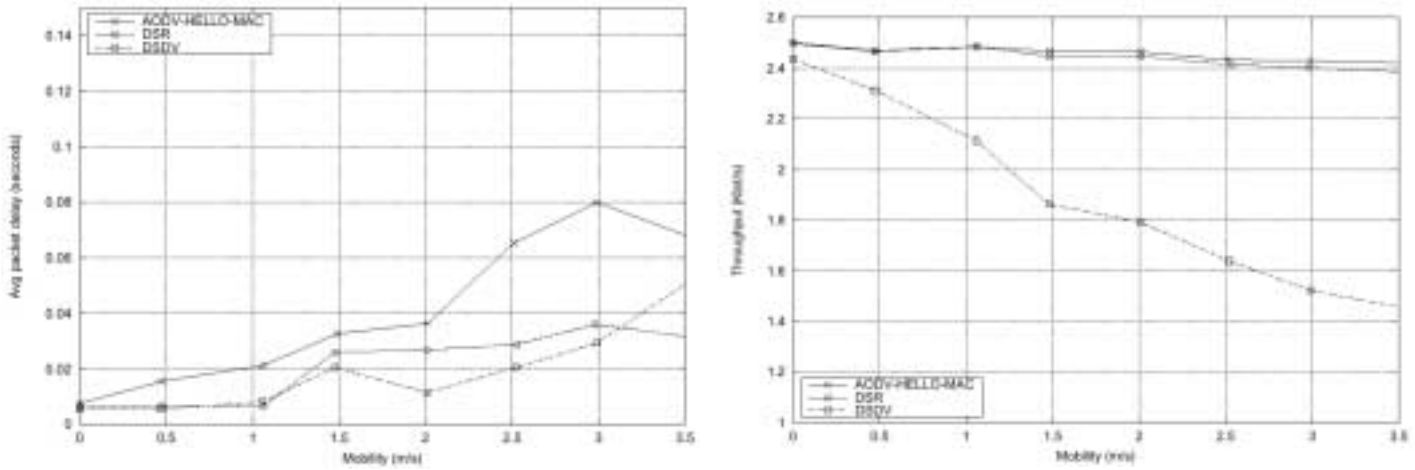


Figure 17 - Délai et débit / mobilité

La figure 18 permet de comparer pour trois protocoles le pourcentage de paquets routés en fonction de l'écart à la valeur optimal de sauts pour des sources CBR fixes à 5 paquets de 64 octets par seconde [4]. Les différences sont faibles, DSDV obtient de meilleurs résultats avec une valeur moyenne d'écart de 0,13 sauts.

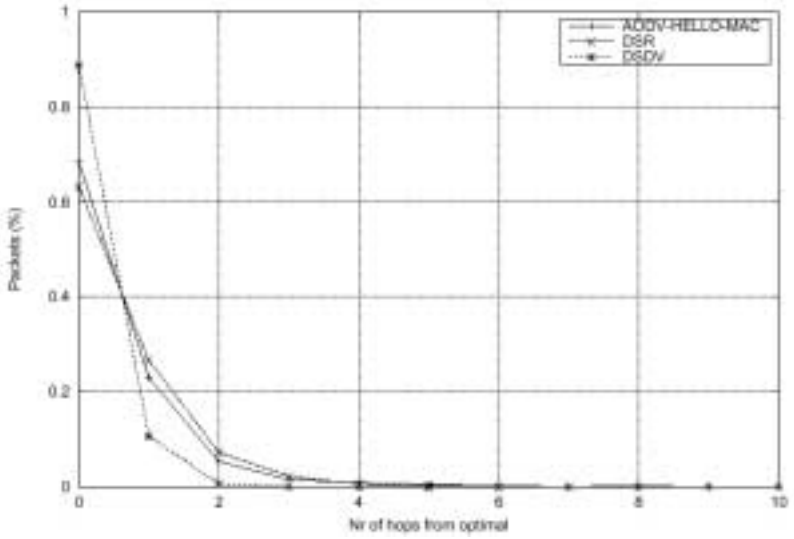


Figure 18 - % paquets routés / nombre de sauts

2.10. Conclusion

Les protocoles purement **proactifs** ne sont pas appropriés pour des réseaux étendus, denses et à forte mobilité, puisque la capacité du réseau est alors majoritairement utilisée pour maintenir à jour les informations de routage dans des tables dont la taille croît linéairement avec le nombre de nœuds. En particulier, DSVD donne des résultats médiocres lorsque la mobilité augmente.

L'intérêt de l'approche **réactive** est la réduction du trafic de paquets de routage, puisque les nœuds ne gardent en mémoire aucune route et ne génèrent de procédure de découverte de route que lorsqu'ils ont des paquets à émettre. Ce type de routage est donc adapté pour des réseaux de topologie très dynamique. DSR et AODV donnent de bons résultats en terme de délai et de débit, quelque soit la mobilité des nœuds. La surcharge occasionnée par le routage à la source (DSR) ou les phases de découverte (AODV) reste comparable. TORA obtient les moins bonnes performances en terme de surcharge. Les protocoles réactifs sont par ailleurs plus difficilement utilisables dans les applications temps réel, puisqu'ils introduisent des délais imprévisibles, et dans les applications multimédia où une certaine qualité de service est indispensable.

L'approche **hybride** combine certains des avantages des deux types mais les problèmes de latence se retrouvent lors de l'échange de données entre des nœuds éloignés dans le réseau.

L'avantage des protocoles **hiérarchiques** est de diminuer le trafic des paquets de routage et la taille des données à conserver, puisqu'elles se limitent au contenu d'une zone primaire. Cependant, la gestion de mobilité est difficile et le maintien des différents niveaux de hiérarchie requiert des algorithmes complexes. De plus, le routage inter-zone par les nœuds périphériques présente de nombreux inconvénients (panne du nœud, point de congestion, consommation importante…).

Il est difficile de conclure sur la prédominance d'un protocole par rapport aux autres et les approches proactives et réactives présentent chacune des avantages suivant les caractéristiques de densité, de mobilité et le type d'applications du réseau ad hoc. De plus, les simulations présentées ont été réalisées avec beaucoup d'hypothèses différentes sur les configurations utilisées et notamment aux niveaux 1 et 2 et ne concernent pas tous les protocoles.

Les problèmes de transmission sont souvent minimisés et les modèles utilisés se basent le plus souvent sur la zone de couverture en sous-estimant l'impact des interférences. Des algorithmes de routage complexes utilisant de nombreux messages de contrôle sont souvent proposés sans faire d'hypothèses sur le nombre effectif de nœuds dans le réseau et la sporadicité du trafic. De plus, les caractéristiques différentes des nœuds (ressources, puissance, énergie) sont rarement prise en compte (toutes les stations n'ont pas les moyens de devenir routeur). Enfin, Les critères de qualité de service proposés par certains concernent seulement les coûts de transmission en terme de nombre de saut.

3. Routage QoS

3.1. Généralités

Le routage QoS peut être introduit de différentes manières :

- à partir de protocoles de routage existants (proactif, réactif, routage à la source ou distribué…), en superposant un protocole capable de différentier les routes suivant les métriques choisis ;
- en développant des protocoles spécifiques, éventuellement inspirés du monde filaire, et conçus à la base pour orienter le routage suivant des contraintes de qualité.

Par ailleurs, les protocoles de routage QoS peuvent être classés suivant le métrique utilisé [16] :

- contrainte de délai ;
- contrainte de bande passante ;
- contrainte de coût (nombre de sauts, ressources sollicitées à chaque nœud, taux d'utilisation des liens…).

Ces métriques sur un chemin $P = i \rightarrow j \rightarrow \ldots k \rightarrow l$ peuvent être définis par les relations :

$$délai(P) = délai(i,j) + \ldots + délai(k,l)$$

$$BP(P) = \min \{BP(i,j),\ldots,BP(k,l)\}$$

$$coût(P) = coût(i,j) + \ldots + coût(k,l)$$

Du fait de la mobilité, la durée de vie limitée des routes doit être prise en compte. Si un nœud se déplace, la QoS prévue risque de ne pas être assurée pendant toute la communication. Si la QoS est garantie tant que les routes restent valides, il est possible de tolérer, en fonction des exigences des applications, des périodes de transition qui correspondent à la réorganisation des chemins et pendant lesquelles le trafic est de type « *best-effort* ». Ces solutions sont de type « *soft QoS* » [22]. A l'inverse, certains protocoles sont basés sur la recherche d'une route maximisant la probabilité de respect des critères de QoS [29].

Quelque soit l'approche ou le métrique, l'estimation des ressources est réalisée en liaison avec les couches inférieures et la réservation en liaison avec les couches supérieures. Quelques-uns des travaux sont résumés ci-dessous.

3.2. Routage QoS sur DSDV

L'extension proposée [17] est destinée au départ à assurer une QoS pour des communications orientées temps réel liées à des applications multimédia de type audio ou vidéo dans un réseau de mobile sans infrastructure. Un circuit virtuel (VC) peut être établi pour transporter ces flux temps réel si la bande passante évaluée sur tout le chemin est suffisante. Dans le cas contraire, les paquets serons transmis en mode datagramme.

Le protocole est basé sur des transmissions de type TDMA. Lors d'une demande de réservation, le protocole évalue la bande passante disponible sur la route principale fournie par DSDV en déterminant le nombre slots TDMA disponibles sur chaque lien tout au long de la route. DSDV étant basé sur le calcul du plus court chemin, le VC correspondra à la route la plus courte avec le nombre minimum requis de slots disponibles tout au long.

Les informations de bande passante sont intégrées aux tables de routage de DSDV et échangées pour calculer la bande passante de bout en bout sur le plus court chemin entre une source et une destination.

Les transmissions sont synchronisées pour l'ensemble du réseau sur des trames contenant un nombre fixe de slots. Ces trames comportent deux phases (figure 19) :

- La phase de contrôle qui a pour rôle de synchroniser les trames et de préciser pour chaque nœud les slots occupés. Chaque station dispose d'un slot de contrôle (1 à N) pour diffuser ces informations à ses voisins.
- La phase de données pour transporter les information de VC ou le trafic datagramme.

A la fin de la phase de contrôle, chaque station doit avoir appris la répartition des slots de la phase de données.

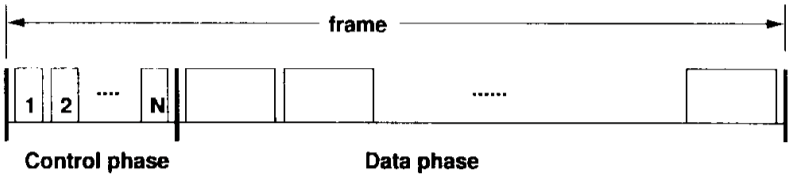


Figure 19 - Structure de la trame

Dans la mesure où seuls les nœuds adjacents peuvent entendre ces information de réservation, les slots libres enregistrés à chaque nœud peuvent être différents. La bande passante sur un lien (*link bandwith*) correspond donc aux slots communs libres entre deux nœud adjacents. Par suite, la bande passante de bout en bout (*path bandwith*) peut être évaluée de proche en proche par un algorithme tenant compte des bandes passantes sur chaque lien et des tables de routage fournies par DSDV.

Dans l'exemple donné figure 20, si le nœud B peut évaluer la bande passante vers A, C peut utiliser cette information et la bande passante sur le lien vers B pour calculer à son tour la bande passante vers A.

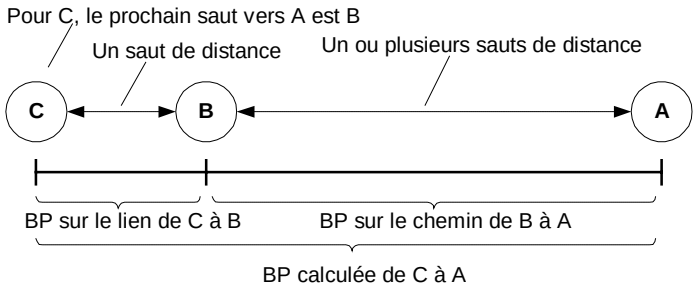


Figure 20 - Calcul de bande passante de bout en bout

Pour évaluer la bande passante sur un lien, il suffit de connaître le nombre de slots disponibles à chaque nœud. Dans l'exemple précédent, si C a les slots libres {1, 3, 4} et B les slots libres {1, 2, 3}, la bande passante sur le lien CB sera {1,3}. Plus généralement : link_BW (P,Q) = free_slot(P) ∩ free_slot(Q). La variable free_slot(X) étant l'ensemble des slots non utilisés par tous les nœuds adjacents à X pour les communications avec X.

La figure 21 illustre l'algorithme utilisé pour calculer la bande passante de bout en bout [25]. La phase de donnée est découpée en 10 slots. La notation « _ » signifie que le slot est réservé et non disponible.

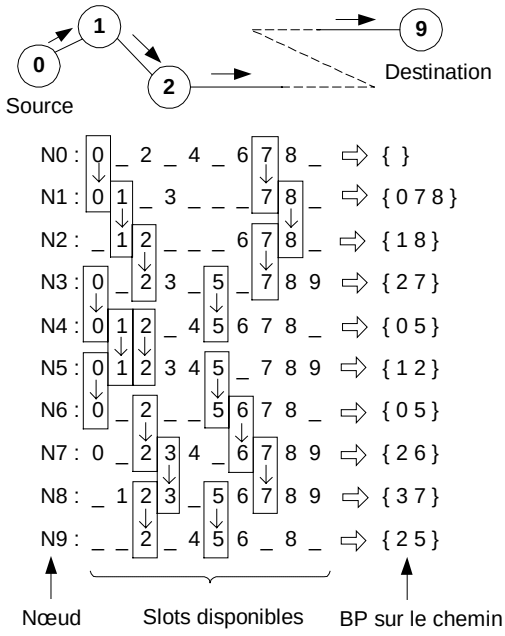


Figure 21 - Exemple de calcul de bande passante

Pour les premiers liens, on obtient :

free_slot(0) ={0,2,4,6,7,8} et par conséquent link_BW (1,0) = path_BW (1,0) = {0,7,8}.

path_BW (2,0) est calculé à partir de link_BW (2,1) et path_BW (1,0) ; on obtient {1,8}, etc.

La QoS possible dans cet exemple correspond à 2 slots.

Après calcul de la bande passante de bout en bout, la réservation est effectuée à partir de la destination, de proche en proche jusqu'à la source. Dans l'exemple, si QoS = 2, le nœud 9 réserve les slots 2 et 5, le nœud 8, les slots 3 et 7, etc. Ces slots sont réservés dans chaque trame (spécifié dans la phase de contrôle) jusqu'à la fin de la session. Une fois les réservations effectuées, le nœud source peut commencer à transmette ses informations sur le VC.

La figure 22 donne les résultats de simulation pour différents types de QoS avec des réservations de 1, 2 ou 4 slots par trame (hybrid QoS correspond à une réservation aléatoire sur une session) . L'environnement prend en compte 20 nœuds mobiles émettant à 4 Mbit/s avec une portée de 400 ft sur une surface de 1000 x 1000 ft. On observe que le nombre de paquets re-routés après une rupture de chemin atteint rapidement 50% quelque soit le nombre de slots réservés. Le débit augmente avec le nombre de slots réservés et varie peu avec la mobilité.

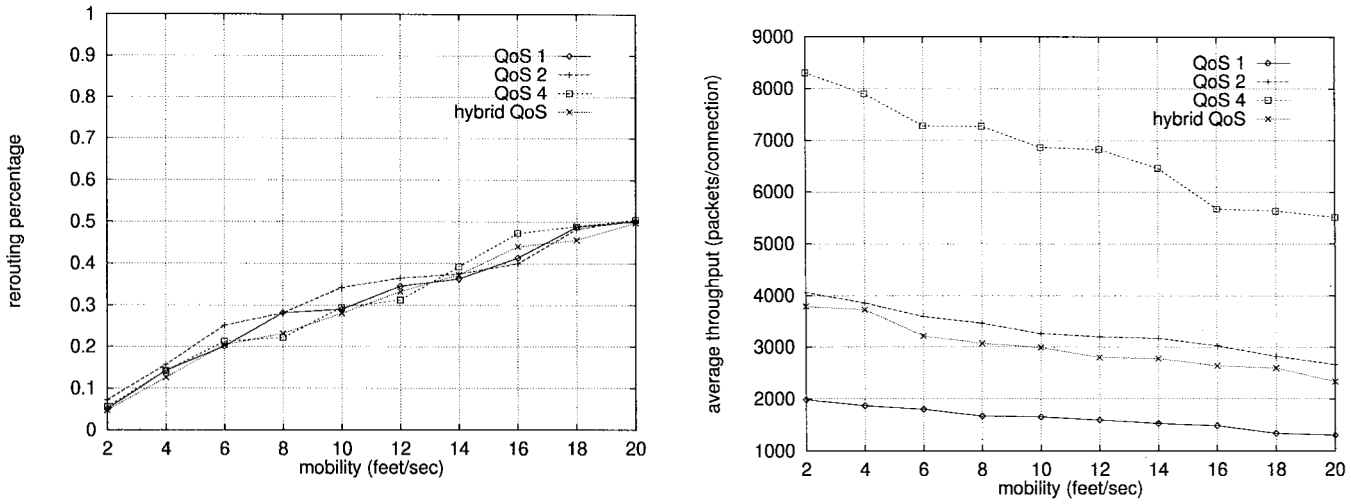


Figure 22 - Paquets re-routés et débit / mobilité

Cette extension basée sur les couches inférieures semble obtenir de bons résultats en ce qui concerne l'évaluation de la bande passante mais l'algorithme utilisé ne tient pas compte de l'évolution des slots libres sur les différents liens (fonction des trafics entrants et de la mobilité). De plus, la route la plus courte n'est pas forcément celle qui présente la meilleure bande passante . Il est donc peu adapté à des réseaux très mobiles présentant de grandes variations de flux et avec des ressources inégales dans les nœuds.

3.3. Routage QoS sur AODV

Cette extension [18] utilise également un métrique de bande passante basé sur la gestion du nombre de slots TDMA à partir des besoins de la source. Dans la mesure où le protocole de routage associé est réactif, les routes avec QoS ne sont établies que sur requête. Le routage QoS va donc simultanément déterminer la route et les slots nécessaires pour chaque lien de la route en fonction de la requête initiale.

Un algorithme de mesure de la bande passante disponible sur la route tracée par AODV est mis en œuvre en même temps que la recherche de route (RREQ).

Chaque nœud est capable de déterminer au fur et à mesure les slots libres pour le nouveau flux.

Les slots libres pour chaque nœud sont évalués en fonction des slots occupés pour émettre ou recevoir avec ses voisins.

On peut ainsi définir l'ensemble des slots libres pour qu'un nœud n_i émettent sans causer d'interférences à ses voisin récepteurs (SRT_i) ainsi que l'ensemble des slots libres pour qu'un nœud n_i reçoive sans subir d'interférences de ses voisins émetteurs (SSR_i). Dans l'exemple illustré figure 23, chaque nœud dispose de 2 slots s_1 et s_2. La transmission courante entre n_1 et n_2 se fait sur le slot1. Les nœuds n_1 et n_2 ne peuvent plus émettre ni recevoir sur le slot 1 ; n_3 ne peut pas recevoir sur s_1 ; n_4 ne peut pas émettre sur s_1.

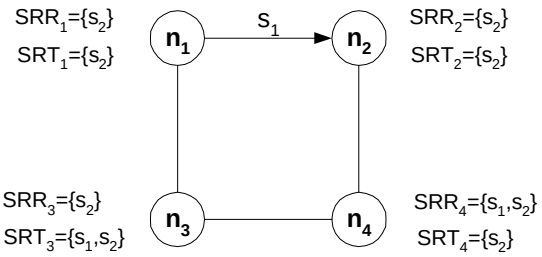


Figure 23 - Exemple de mesure de slots TDMA

A partir des ensembles SRT_i et SRR_i du nœud n_i, il est possible d'évaluer le nombre de slots disponibles sur le lien vers le nœud suivant n_{i+1} de la route et par suite, pour chaque lien de la route.

La bande passante de bout en bout est ensuite calculée de proche en proche suivant un algorithme itératif prenant en compte les 3 liens les plus proches d'une destination (considéré comme suffisant pour éviter les problèmes d'interférence) et les bandes passantes calculées des 2 liens précédents.

Dans l'exemple de la figure 24, la BP entre n_3 et n_2 (égale à s_3, s_6) est calculée à partir des slots libres sur les liens n_5n_4, n_4n_3, n_3n_2 et à partir de la BP calculée sur les liens n_5n_4 et n_4n_3.

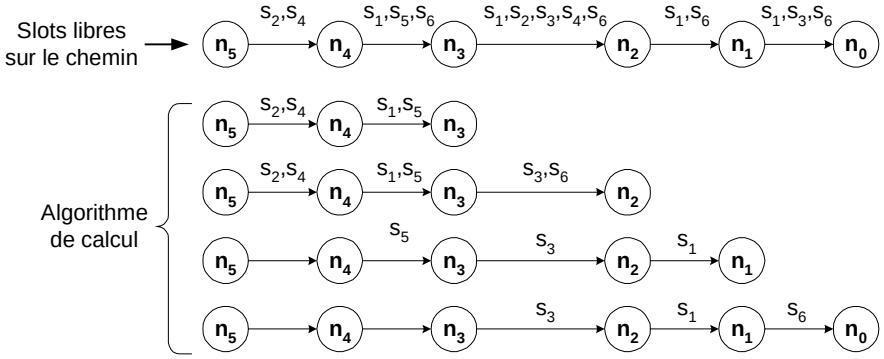


Figure 24 - Exemple de calcul de bande passante

Les paquets RREQ du protocole AODV sont enrichis avec les informations de bande passante liées à l'algorithme utilisé.

Après évaluation de la bande passante de bout en bout, les paquets de réponse (RREP) remontent et réservent les slots jusqu'à la source.

La figure 25 donne les résultats de simulation en comparant QoS et BE (*Best Effort*) pour différentes mobilités (v = 0, 5, 10 m/s). L'extension améliore les performances par rapport à AODV *Best Effort*. Les meilleurs résultats sont obtenus pour des petits réseaux avec des routes courtes et une faible mobilité.

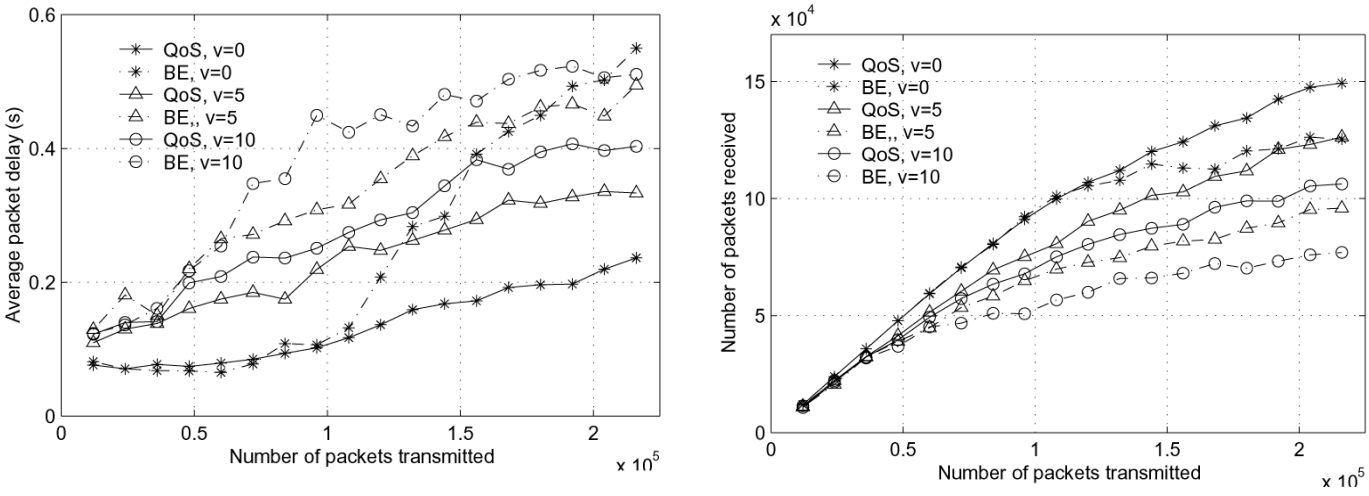


Figure 25 - Délai et nombre de paquets reçus / nombre de paquets transmis

3.4. CEDAR (*Core Extraction Distributed Ad hoc Routing algorithm*)

Le protocole CEDAR [29] repose sur l'élection dynamique par les nœuds d'un cœur de réseau stable formé par des nœuds dominants d'un point de vue topologique. L'élection des nœuds du cœur est réalisée à l'aide de paquets balises (*beacons*) suivant un algorithme assurant que tous les nœuds sont soit un dominant soit voisin d'un dominant avec un minimum de dominants (voir figure 26).

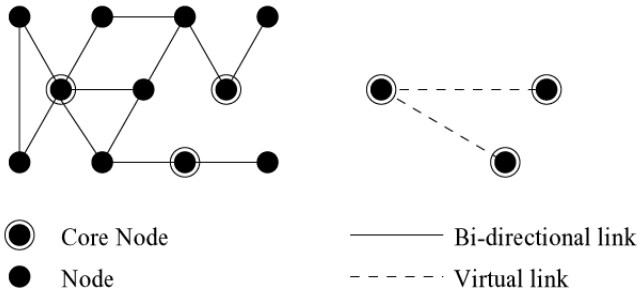


Figure 26 - Exemple d'ensemble de nœuds du cœur

Le rôle des nœuds du cœur est de collecter et propager les informations sur la bande passante disponible dans les liens.

Chaque nœud du cœur maintient ainsi une table sur sa topologie locale ainsi que sur l'état des liens éloignés en terme de stabilité et de bande passante minimum.

Une augmentation ou une diminution de BP (en fonction de seuils prédéfinis) mesurée sur un lien par un nœud doit être signalée à son dominant. Ce dernier effectue alors une diffusion vers le cœur en indiquant le lien concerné et le sens de variation.

Les nœud dominants doivent également assurer le routage suivant un protocole réactif.

La route avec QoS est établie en 2 temps (figure 27) :

- découverte par les nœuds dominants d'une route cœur à l'aide du protocole de routage associé ;
- recherche d'une route avec QoS basée sur la route cœur.

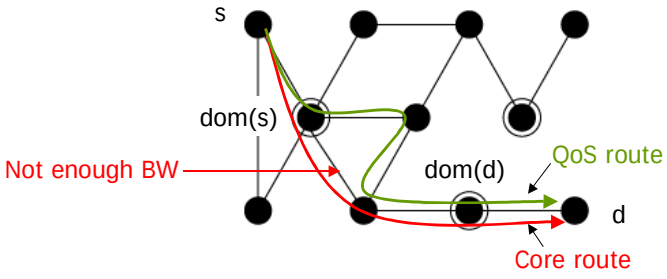


Figure 27 - Exemple de recherche d'une route QoS

En cas de rupture de lien, deux mécanismes sont mis en œuvre en parallèle :

recherche dynamique d'une route temporaire possible au voisinage du point de rupture ;

- information de rupture transmise à la source qui relance une recherche complète de route avec QoS.

CEDAR repose sur l'utilisation de protocoles d'accès au médium de type CSMA/CA minimisant les risques de collision et qui doivent être capables d'estimer la bande passante disponible sur les liens. Les routes obtenues sont optimisées en terme de bande passante et de nombre de sauts mais en cas de rupture du cœur, le routage est suspendu pendant une période transitoire ce qui peut entraîner des pertes de paquets.

3.5. TBP *(Ticket Based Probing)*

Compte tenu du coût d'accès au médium élevé dans les réseaux ad-hoc, la recherche de routes par inondation (*flooding*) peut devenir très coûteuse. Pour éviter ce problème, la diffusion des requêtes de découverte de route par TBP [16] est limitée en nombre.

Chaque nœud maintient un état local (délai, BP, coût…) pour tous les liens vers ses voisins immédiats grâce à la transmission périodique de paquets de signalisation. Lors d'une recherche de route, la source émet des paquets de sonde (*probe*) comportant un nombre limité N_0 de tickets, un ticket correspond à une route cherchée et une sonde comporte au moins un ticket (le nombre de routes est limité par le nombre de tickets). Le choix de N_0 à la source est basé sur les contraintes de QoS (délai ou BP) et les informations d'état locales (pour un délai demandé important, un seul ticket peut suffire). Les nœud intermédiaires propagent les sondes en répartissant les tickets suivant leurs états locaux (pour une contrainte de délai, un nœud enverra davantage de tickets sur un lien rapide). D'une manière générale, plus un flux de données aura de contraintes, plus on associera de tickets à la demande correspondante.

Dans l'exemple de la figure 28, deux sondes p_1 et p_2 sont envoyées à partir de s. La première comporte un ticket , la deuxième deux. Au nœud j, la sonde p_2 est dédoublée en deux sondes p_3 et p_4 comportant chacune un ticket. Il ne peut pas y avoir plus de trois sondes à la fois et trois chemins sont trouvés : s → i → t ; s → j → t et s → j → k → t.

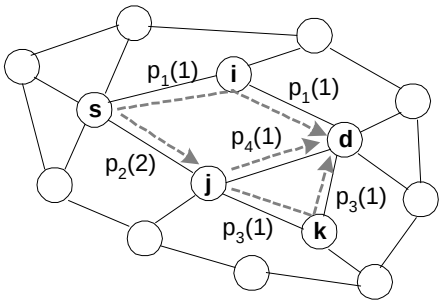


Figure 28 - Principe de limitation des diffusions

La route est mémorisée dans les sondes et après choix d'une route primaire par la destination (celle de meilleure QoS), un message de confirmation remonte à la source pour réservation.

A partir de cette approche générale, les deux principales contraintes de QoS peuvent être satisfaites :

- établir des routes, les plus proches de l'optimal possible, de moindre coût avec des contraintes de délai ;
- établir des routes de moindre coût avec des contraintes de bande passante.

Afin d'augmenter la probabilité de trouver une route, on utilise deux types de tickets : les tickets jaunes permettent de rechercher des chemins respectant la contrainte imposée et les tickets verts permettent d'obtenir des solutions de faible coût.

Contrainte de délai

Dans ce cas, le nombre de tickets jaunes Y_0 est déterminé d'après le délai requis D. Si le délai est très grand et peut être assuré, un seul ticket jaune est suffisant. Si D est trop faible pour être assuré, aucun ticket jaune n'est généré et la connexion est rejetée. Dans les autres cas, Y_0 est choisi supérieur à 1.

Le nombre de tickets verts G_0 ($N_0 = Y_0 + G_0$) est aussi déterminé selon D. Pour un délai très faible, aucun ticket vert n'est créé. Si D est très grand, un seul ticket vert suffit. Quand la contrainte de délai diminue, elle est de plus en plus difficile à satisfaire. Le nombre de tickets verts produits doit donc diminuer pour privilégier les tickets jaunes qui favorisent la contrainte de QoS. Quand la contrainte de délai croît, elle est facilement réalisable : le nombre de tickets verts est plus important que celui de tickets jaunes.

La figure 29 illustre la répartition des tickets jaunes et verts suivant la valeur du délai. Si s et t sont respectivement la source et la destination, $D_s(t) + \Delta D_s(t)$ correspond au plus grand délai de bout en bout.

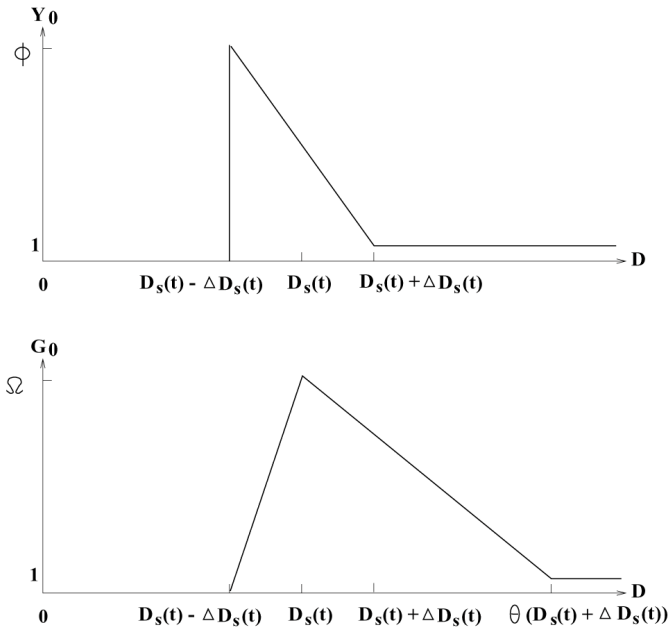


Figure 29 - Nombre de tickets jaunes et verts suivant le délai

Une fois le nombre de tickets à la source choisi, le deuxième problème est la répartition des tickets par les nœuds intermédiaires. Une sonde enregistre au fur et à mesure les délais cumulés. Le nœud qui reçoit la sonde choisi parmi ses voisins ceux qui assureront un délai de bout en bout respectant la contrainte (tickets jaunes) et ceux qui présentent le meilleur coût (tickets verts) vers la destination. La répartition des tickets peut être effectuée avec les règles suivantes :

- les sondes émises sur les liens de plus faible délai doivent avoir davantage de tickets jaunes ;
- les sondes émises vers des direction de plus faible coût doivent avoir davantage de tickets verts.

La route est mémorisée au fur et à mesure dans les sondes et la recherche de route est terminée lorsque toutes les sondes et tous les tickets Y_0 et G_0 sont arrivées à destination (les tickets non valides doivent tout de même êtres transmis par les nœuds intermédiaires). Suivant le nombre de tickets jaunes reçus, la destination choisi comme route primaire celle de meilleure coût satisfaisant la contrainte de délai (le coût est enregistré dans toutes les sondes). En l'absence de ticket jaune valide, les tickets verts sont retenus et la route de meilleure coût est choisie. Les routes secondaires de moindre QoS pourront être utilisées en cas de rupture de lien. Un message de confirmation remonte à la source et effectue les réservations.

Contrainte de bande passante

Le principe de répartition des tickets jaunes et verts à la source est sensiblement le même. Les nombres Y_0 et G_0 évoluent entre 0 pour une bande passante B requise trop importante et une valeur maximale (Φ ou Ω) en passant par la valeur 1 pour une faible bande passante requise et facilement réalisable (figure 30).

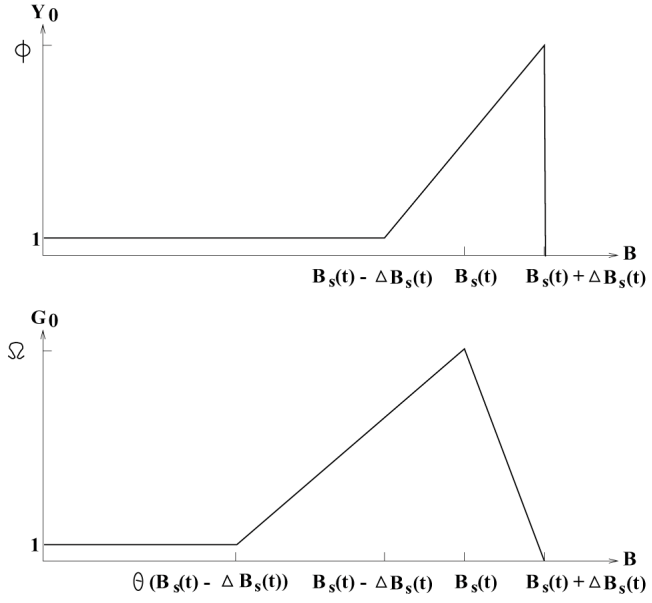


Figure 30 - Nombre de tickets jaunes et verts suivant la BP

La répartition des tickets jaunes et verts par les nœuds intermédiaires suit des règles du même type :

- les sondes émises sur les liens de plus large bande passante doivent avoir davantage de tickets jaunes ;
- les sondes émises vers des direction de plus faible coût doivent avoir davantage de tickets verts.

La figure 31 donne les résultats de simulation en comparant TBP à un algorithme d'inondation (*flooding*) et à un algorithme de recherche de plus court chemin (*Shortest Path*). La simulation est conçue avec 40 nœuds placés dans un carre de 15x15 m avec des portées de 3 m.

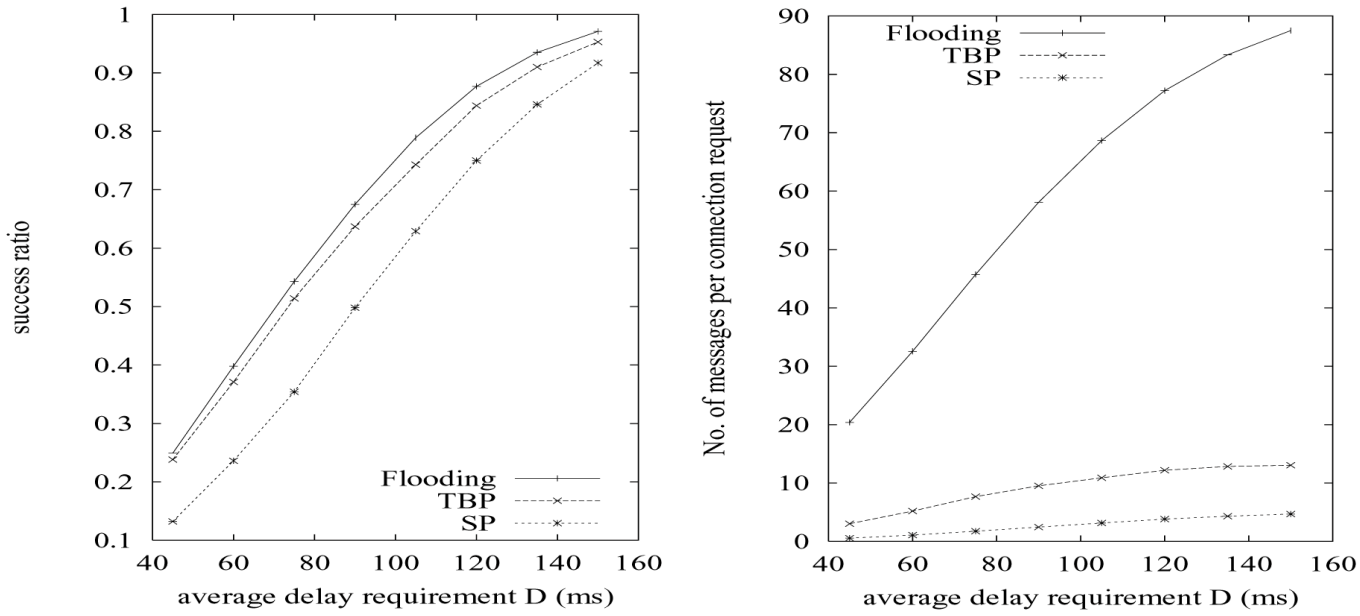


Figure 31 - % Succès et nombre de messages par requête / délai moyen requis

Malgré le fait que les nœuds ne connaissent que leur voisinage immédiat, TBP est efficace puisqu'il permet de trouver des routes avec une probabilité proche des algorithmes basés sur l'inondation et meilleure que des algorithmes de type SP. Il permet en outre de trouver des routes de plus faible coût que ces deux types d'algorithmes.

Ce protocole de routage QoS a été conçu pour des réseaux dans lesquels la mobilité est suffisamment faible pour ne pas poser de réel problème (scénario de type salle de conférence). La durée de vie des routes doit être grande devant le temps nécessaire à l'établissement ou à la restauration d'une route. Par ailleurs, la couche MAC et les techniques d'évaluation des ressources ne sont pas définis.

3.6. Comparaison des performances

Le tableau ci-dessous résume les caractéristiques et les performances des solutions étudiées pour fournir de la qualité de service au niveau du routage dans les réseaux ad hoc.

	QoS sur DSDV	QoS sur AODV	CEDAR	TBP
Approche	Proactive	Réactive	Réactive	Réactive
Maintenance d'une table	Oui	Oui	Oui	Oui
Multipath	Non	Non	Non	Oui
Couche liaison	TDMA	TDMA	CSMA/CA	Non défini
Métrique	BP	BP	BP	Délai, BP ou coût
Evaluation du métrique	Oui	Oui	Non	Non
Surcharge QoS	Forte	Moyenne	Forte	Faible
Mobilité	Faible	Faible	Moyenne	Faible
Densité	Moyenne	Moyenne	Moyenne	Faible

Tableau 3 - Comparaison des solutions de QoS

Les trois premières solutions sont basées sur l'évaluation et la réservation de ressources en liaison avec la couche MAC et font un certain nombre d'hypothèses sur celles-ci (multiplexage TDMA, méthode d'accès CSMA/CA). Seule TBP propose une solution de QoS multicritère mais ne donne aucune indication sur la mesure des ces métriques. Par ailleurs toutes ces propositions sont prévues pour des réseaux peu denses et de faible mobilité.

3.7. Conclusion

Au vue de ces études et si l'on considère que le routage QoS doit nécessairement s'appuyer sur la couche MAC [32], deux approches sont possibles.

- L'évaluation des ressources est réalisée par la couche MAC, la route est ensuite tracée en fonction des métriques évalués. Cette approche est bien adaptée à des méthodes d'accès déterministes du type TDMA pour lesquelles la bande passante disponible correspond à un nombre relativement constant de slots de temps libres pendant tout une transmission [17, 18]. La réservation qui correspond également à un nombre de slots de temps affecté à chaque station ou chaque lien de la route tracée est aussi effectuée en liaison avec la couche MAC.

  Pour des méthodes d'accès aléatoire type MACA ou CSMA/CA, l'évaluation des ressources peut toutefois être réalisée par l'échange périodique de sondes (*beacons*) chargées de collecter et de propager des informations de délai ou de bande passante sur un lien ou une route. Les 2 inconvénients majeurs dans ce cas étant la surcharge au niveau MAC et la validité dans le temps de ces estimations.

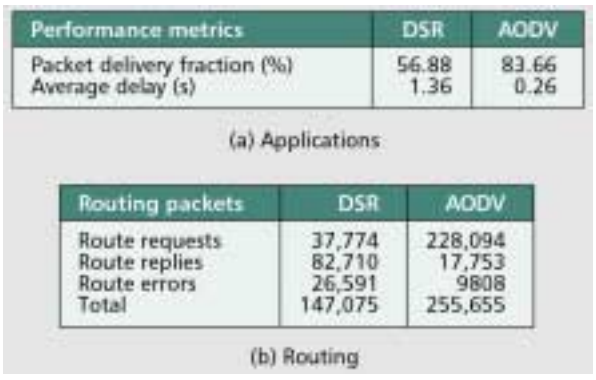
- Un mécanisme de différentiation est mis en œuvre au niveau de la couche MAC [30, 31] pour affecter des priorités aux stations de la route choisie par le protocole de routage (variation du nombre de timeslots, du DIFS ou de la longueur des trames suivant la priorité). Cette approche est adaptée aux méthodes d'accès aléatoire type MACA ou CSMA/CA pour lesquelles une évaluation préalable des ressources disponibles est complexe. Dans ce cas, le routage QoS n'est plus seulement envisagé comme la recherche d'une route avec contraintes, mais plus globalement, dans le cadre d'un modèle QoS (voir § 1.3). Par ailleurs lorsque plusieurs flux de QoS différentes doivent circuler dans le réseau ad hoc, la mise en place de priorités distinctes sur un nœud routeur peut devenir impossible.

Dans le mesure où la norme 802.11 et ses variantes semble s'imposer dans les années à venir, une solution de routage QoS adaptée à la fonction DCF (voir annexe 802.11) sera envisagée ; l'évaluation des ressources sera résolue au niveau MAC.

Par ailleurs, il est nécessaire de tenir compte des ressources inégales des nœuds (portables, PDA…) et des caractéristiques très variables des réseaux ad hoc. La solution qui semble la plus performante est d'utiliser et d'étendre un protocole de routage réactif capable de s'adapter à la densité, à la mobilité, au trafic et aux ressources des stations.

Une étude comparative très complète [33] des deux principaux protocoles réactifs DSR et AODV implantés sur un réseau ad hoc utilisant le protocole 802.11 avec un contrôle RTS/CTS met en évidence leurs caractéristiques.

La figure 32 montre les résultats d'une simulation réalisée pour un réseau de 100 nœuds en mobilité constante à des vitesses variant entre 0 et 20 m/s. Le trafic est généré par 40 sources CBR émettant deux paquets de 512 octets par seconde.



Performance metrics	DSR	AODV
Packet delivery fraction (%)	56.88	83.66
Average delay (s)	1.36	0.26

(a) Applications

Routing packets	DSR	AODV
Route requests	37,774	228,094
Route replies	82,710	17,753
Route errors	26,591	9808
Total	147,075	255,655

(b) Routing

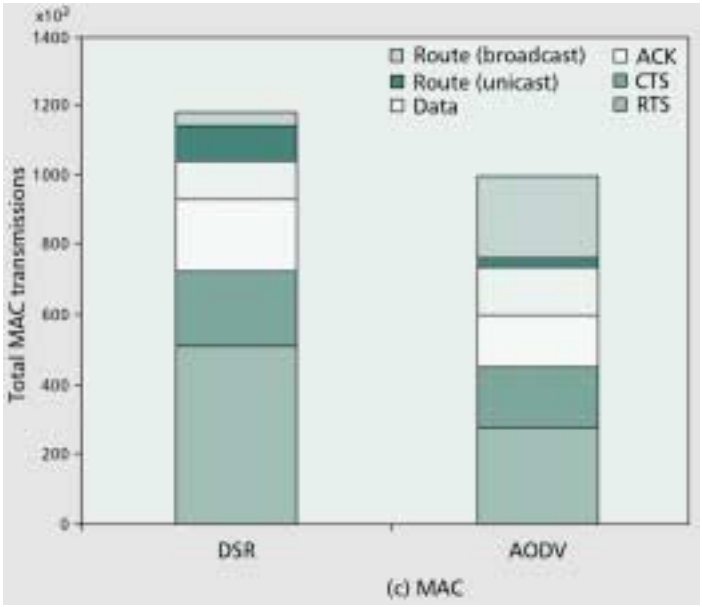


Figure 32 - Performances des protocoles DSR et AODV

Les résultats montrent que pour les applications orientées métriques comme le délai ou la bande passante, DSR donne de meilleures performances pour des situations moins contraignantes en terme de densité, de trafic et de mobilité. En revanche, AODV est plus performant dès que les contraintes augmentent (davantage de charge, plus grande mobilité des stations). Quelques soient les situations, DSR génère constamment moins de surcharge de routage qu'AODV.

Par ailleurs, les interactions entre le routage et la couche MAC affectent de manière significative les performances. Même si DSR génère globalement moins de surcharge de routage, il utilise davantage de paquets unicast et par conséquent de trames MAC. Cette observation confirme l'importance d'étudier les interactions entre les différents niveaux pour apporter la QoS.

En définitive, le protocole AODV sera retenu pour ses meilleures capacités d'adaptabilité aux différentes situations de réseau ad hoc, notamment les plus contraignantes. De plus, AODV gère le nombre de sauts et intègre des possibilités de routage multicast, ce qui présente des avantages en termes de QoS.

4. Extension QoS sur AODV pour 802.11

4.1. Principe

L'extension QoS envisagée utilise les deux métriques délai et bande passante qui peuvent être définis sur un chemin `P = i→j→…k→l` par les relations :

`délai(P) = délai(i,j) + … + délai(k,l)`

$BP(P) = \min \{BP(i,j),\ldots,BP(k,l)\}$

L'estimation du délai est réalisée successivement, pour chaque nœud de la route tracée, au niveau routage et non au niveau MAC. La bande passante est évaluée également de proche en proche sur la route, pour chaque paire source/destination.

La route avec QoS est donc tracée de nœud en nœud suivant le protocole AODV. Pour chaque nœud traversé, un test est effectué pour savoir si les exigences de délai maximum ou de bande passante minimum pourront être satisfaites. Dans la négative, si le délai cumulé est déjà trop long au niveau d'un nœud intermédiaire ou la bande passante sur un lien traversé trop faible, la recherche de route sera interrompue.

Le routage QoS reste donc réactif et utilise des extensions sur les paquets de requête et de réponse.

4.2. Estimation du délai

L'estimation envisagée est basée sur l'aspect réactif du protocole de routage choisi. Il ne s'agit donc pas d'un calcul préalable du délai de bout en bout mais d'un algorithme basé sur l'évaluation successive des délais de transmission entre les nœuds d'une route. Plus précisément, l'estimation utilise l'un des paramètres du protocole AODV, le NODE_TRAVERSAL_TIME considéré à la base comme une constante [6]. Ici, le NTT devient une estimation du temps de traversé moyen d'un paquet pour un saut et inclut le délai de transmission sur le lien et le temps de traitement dans le nœud (délais dans les files, temps d'interruption des processus…).

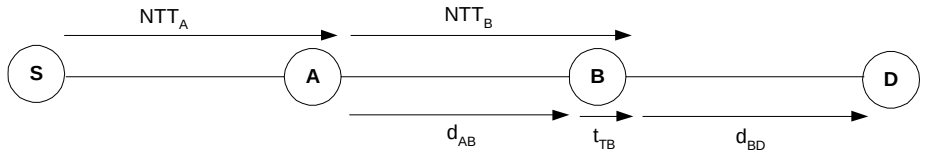


Figure 33 - Evaluations successives des NTT

Sur une réseau 802.11, le NTT est donc fonction de la durée de la contention et des éventuels retransmissions après collision ; des durées d'émission des différentes trames (RTS, CTS, données, ACK) ; des temps inter trames (DIFS, SIFS) ; des différents temps de propagation et du temps de traitement dans le nœud .

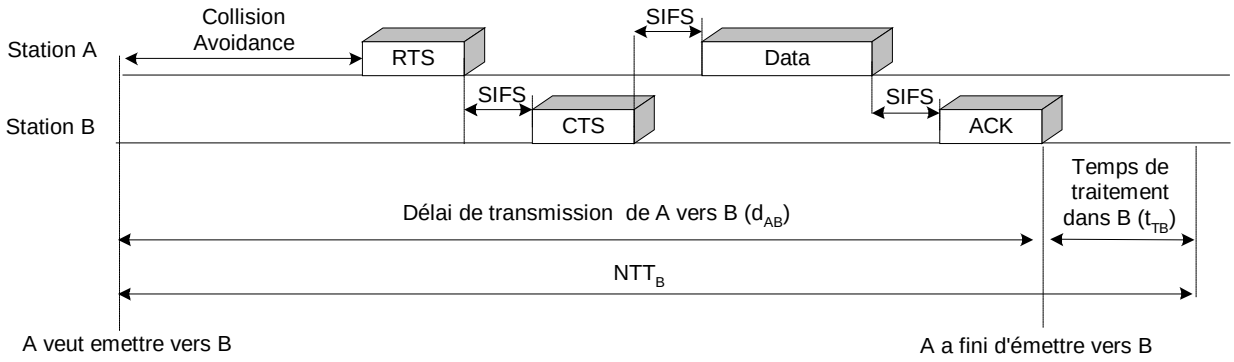


Figure 34 - Décomposition du NTT

Pour le noeud B :

$$NTT_B = d_{AB} + t_{TB}$$

Et le délai au niveau MAC si aucune collision ne survient :

$$d_{AB} = t_{CA} + t_{RTS} + t_{CTS} + t_{Data} + t_{ACK} + 3t_{SIFS} + 3\tau$$ (τ étant le temps de propagation sur le lien).

Le temps de traitement dans le nœud ($t_{T\,B}$) peut être considéré comme une constante spécifique à chaque nœud. Le délai de transmission entre les deux nœuds (d_{AB}) correspond au temps entre l'instant où le paquet est transmis à la couche MAC pour être émis par le nœud source et l'instant d'émission de l'acquittement par le nœud destination :

$$d_{AB} = T_{ACK} - T_{transmission}$$

Si les horloges sont synchronisées entre les nœuds, la mesure de d_{AB} peut être réalisée par la transmission de paquets de requête RREQ et de réponse RREP munis d'extensions temporelles (*Timestamp*) suivant un format prévu par AODV.

La synchronisation des horloges peut être obtenue dans un phase initiale par échange de sondes au niveau MAC ou routage suivant un protocole de type NTP.

Le délai mesuré d_{AB} est lié à la taille des paquets, une correction devra être effectuée pour tenir compte d'une taille moyenne et non de la longueur des paquets RREQ ou RREP utilisés pour la mesure. Pour des paquets RREQ et RREP de 24 octets et des paquets de données d'une taille moyenne de 100 octets (taille liée au paramètre *Fragmentation Thresold* de la norme 802.11) transmis à 11 Mbit/s sur le réseau 802.11, la correction peut être calculée :

$$d'_{AB} = d_{AB} + \frac{(100-24)\times 8}{11.10^6} \approx d_{AB} + 55\mu s$$

Par ailleurs, dans la mesure où les délais sur les routes dépendent d'évènements aléatoires (déplacement, arrivées, extinction, variation des flux et des trafics, etc.) intervenants à différents niveaux du réseau, la variance des délais de nœud à nœud peut être importante. Pour tenir compte de ces variations dans le temps, les deux méthodes courantes [34] sont le calcul d'une moyenne sur une fenêtre de taille fixe ou l'utilisation des mesures précédentes avec une moyenne pondérée par un facteur d'oubli (*exponential forgetting*). Pour limiter la surcharge, nous retiendrons la seconde méthode, ce qui donne pour un délai entre les nœuds A et B :

$$d_{AB}(t) = (1-\lambda)\sum_{k=0}^{\infty} \lambda^k . d_{AB}(t-k)$$

Où $\lambda \in [0,1]$ est le facteur d'oubli.

Une approximation à l'ordre 1 donne :

$$D_{AB}(t) = (1-\lambda) d_{AB}(t) + (1-\lambda)\lambda D_{AB}(t-1)$$

où $d_{AB}(t)$ est le délai de transmission entre A et B mesuré à l'instant t.

Comme indiqué précédemment, la mesure de d_{AB} (et par suite, le calcul de D_{AB}) peut être réalisée par émission de sondes au niveau MAC intégrant une étiquette de temps.

La solution retenue ici est basée sur la transmission de paquets munis d'extensions temporelles (voir format au §4.4). Le protocole AODV prévoit en effet, à partir de la révision 10, la possibilité d'ajouter une extension « *Timestamp* » aux paquets de requête RREQ et de réponse RREP [6].

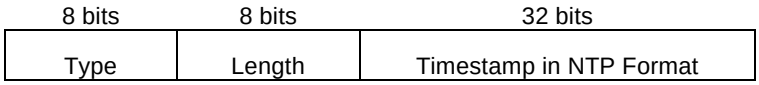


Figure 35 - Format de l'extension Timestamp

Type : 3

Length : 6

Timestamp : réel codé sur 64 bits avec 32 bits pour la partie entière et 32 bits pour la partie fractionnaire et donnant le nombre relatif de secondes depuis le 1/1/1900 (voire format NTP dans RFC 2030).

Pour assurer la mise à jour fréquente du NTT, tous les paquets RREQ et RREP (munis ou non d'extensions QoS) porteront une extension « Timestamp ». Lorsque des paquets de requête ou de réponse arriveront sur les nœuds d'une route QoS déjà tracée, il seront susceptibles de modifier le NTT, ce qui justifiera la mise à jour du NTT des nœuds concernés.

4.3. Estimation de la bande passante

Comme précédemment, pour garder le caractère réactif du protocole de routage, la bande passante sera évaluée successivement sur chacun des liens de la route et non de bout en bout.

Cette évaluation sur un lien entre une source et une destination peut s'exprimer sous la forme [36] :

$$BP_{disponible} = (1-u) \times Débit_{sur\ le\ lien}$$

u étant le taux d'utilisation du lien.

Pour calculer la bande passante disponible au niveau d'un nœud, il est donc nécessaire d'évaluer dans un premier temps le débit sur le lien. Une première évaluation peut être réalisée simplement par émission de paquet et mesure du temps correspondant [36]:

$$Débit_{paquet} = \frac{S}{T_{ACK} - T_{transmission}}$$

S étant la taille du paquet, $T_{transmission}$ l'instant d'émission du paquet relevé au niveau réseau et T_{ACK} l'instant de réception de l'acquittement provenant du nœud destination.

Ce calcul tient compte de la durée de la contention ; des durées de transmission du paquets de données et des différents paquets de contrôle (RTS, CTS, ACK) ; des temps inter-trame (DIFS, SIFS) ; des délai de propagation et du délai dans la file de niveau MAC. Il peut s'exprimer sous la forme :

$$Débit = \frac{S}{t_q + t_s + t_{CA} + t_{overhead}}$$

t_q étant le délai dans la file de niveau MAC, t_s le temps de transmission des S bits et $t_{overhead}$ regroupant les différentes durées liées au contrôle et à la propagation.

Cette relation montre les inconvénients de cette évaluation :

- dépendance de la taille des paquets ;
- variance importante due au caractère aléatoire de la transmission ;
- délai dans la file devant être évalué suivant que l'instant de transmission est référencé au niveau MAC ou réseau.

Pour limiter l'influence de la taille des paquets, une solution simple consiste à soustraire du délai une constante liée à la surcharge des paquets :

$$c \approx t_{overhead} \approx t_{RTS} + t_{CTS} + t_{ACK} + 3t_{SIFS} + 3\tau$$

Cette constante peut-être calculée en tenant compte des caractéristiques de la liaison 802.11.

La simulation réalisée dans [36] montre que le débit mesuré, après correction par C, est peu influencé par la taille des paquets. Les mesures sont réalisées pour C=1200 µs, avec des paquets sont émis toutes les 20 s sur un lien à 2 Mbits/s.

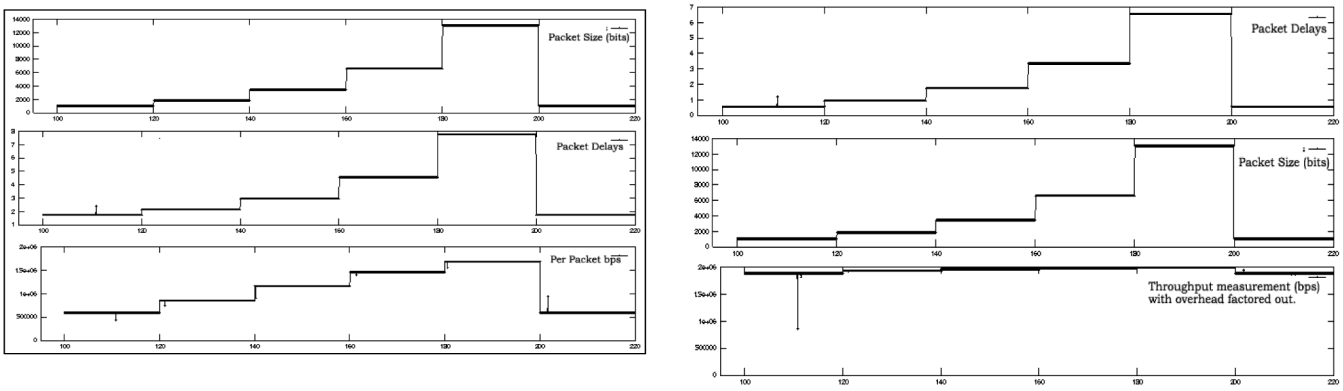


Figure 36 - Délai et Débit / taille des paquets avant et après correction

Pour limiter l'aspect aléatoire de la mesure dans le cas d'une mesure du débit par paquet, une fenêtre d'émission de n paquets peut être utilisée. La mesure du débit moyen pour les n paquets émis, de la durée de la fenêtre (*windows duration*), du temps d'inactivité (*idle time*) pour les n émissions sur le lien permettent ainsi d'évaluer le taux d'utilisation et par suite la bande passante disponible :

$$BP_{\text{disponible}} = \frac{\text{temps d'inactivité}}{\text{durée de la fenêtre}} \times \text{débit mesuré}$$

La figure 37 met en évidence la variance de la mesure du débit par paquet et montre les résultat d'un simulation dans laquelle un filtrage par fenêtre de 32 paquets est réalisé sur un réseau 802.11 à 2 Mbit/s [36].

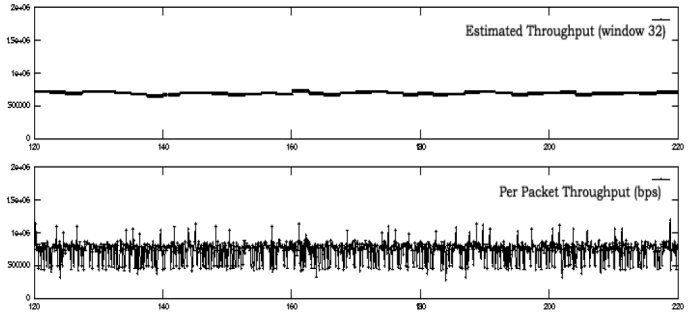


Figure 37 - Estimation du débit mesuré par paquet et par fenêtre

Ici également, la mise en place de cette fenêtre d'émission est réalisée au niveau réseau, en liaison avec le protocole AODV.

L'un des problèmes pour la mesure du débit est de prendre en compte les délai dans la file entre le niveau réseau et le niveau liaison. Les figures suivantes illustrent ce calcul de délai suivant que la transmission à lieu immédiatement ou est différée dans la file MAC.

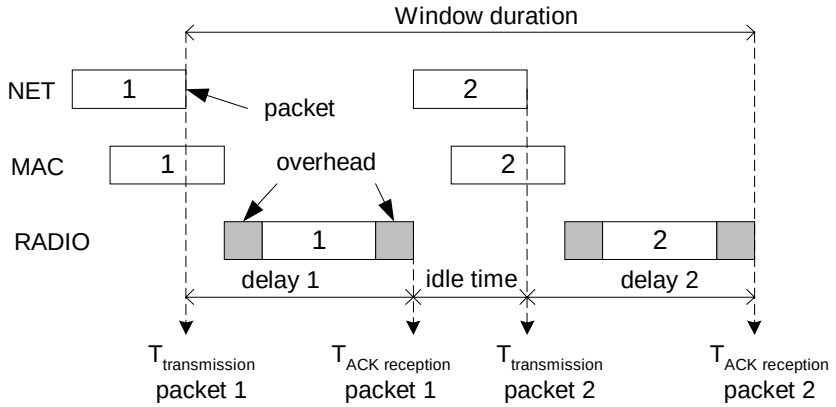


Figure 38 - Délai de transmission sans attente dans la file

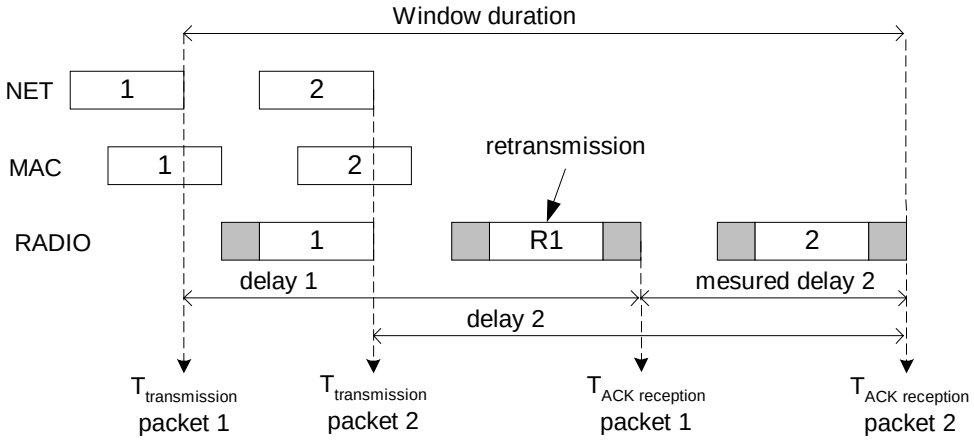


Figure 39 - Délai de transmission avec attente dans la file

Pour chaque requête QoS par l'intermédiaire d'un paquet RREQ, une évaluation de BP disponible au niveau du nœud sera initiée (voir §4.4). L'inconvénient principal de cette mesure est la surcharge occasionnée par l'émission des plusieurs paquets dans une fenêtre. Un compromis entre la précision et la surcharge devra être trouvé pour le dimensionnement de cette fenêtre en nombre de paquets. Cette taille sera également liée aux caractéristiques du réseau ad hoc (charge, mobilité, nombre de requêtes QoS…).

4.4. Routage QoS

Cette extension pour AODV est basée sur la possibilité d'extension prévue par AODV pour ses principaux paquets RREP et RREQ [6].

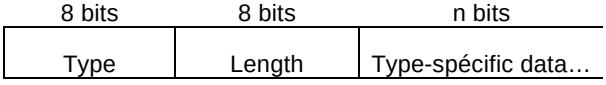


Figure 40 - Format des extensions AODV

Les paquets RREQ peuvent inclure des requête QoS de 2 types :

- délai maximum ;
- bande passante minimum.

Les mêmes extensions sur les paquet RREP indiquent en retour si la requête QoS peut être satisfaite.

Après établissement de la route QoS, les nœuds qui détectent une perte de la QoS (délai augmenté ou BP diminuée) peuvent générer un paquet spécifique RREP au nœud qui à émis la requête QoS.

Extensions pour la table de routage

Les champs suivant sont ajoutés à chaque entrée de route correspondant à chaque destination :

- délai maximum ;
- bande passante minimum disponible ;
- liste des sources demandant des garanties de délai (avec le délai requis) ;
- liste des sources demandant des garanties de BP (avec la BP requise).

Extension QoS délai

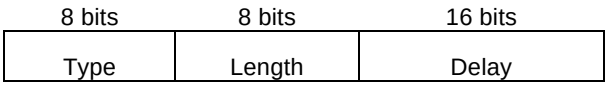


Figure 41 - Format de l'extension QoS délai

Type : 6

Length : 2

Delay : deux significations suivant le type de paquet

- pour un paquet RREQ, il indique le nombre maximum de secondes autorisés pour un transmission entre la source (ou un nœud intermédiaire qui propage le RREQ) et la destination ;
- pour un paquet RREP, il donne une estimation du délai cumulé entre un nœud intermédiaire qui propage le RREP et la destination.

Un nœud avec des contraintes de délai maximum transmet donc un paquet RREQ avec une extension QoS « Delay » (ainsi qu'une extension « Timestamp » permettant la mesure du NODE_TRAVERSAL_TIME, voir §4.2). Avant de propager le paquet RREQ, un nœud intermédiaire compare son NODE_TRAVERSAL_TIME au délai restant indiqué dans l'extension.

Si le délai est inférieur, le paquet est écarté et le processus s'interrompt.

Si le délai est supérieur, le nœud soustrait son NTT du délai fourni dans l'extension et continu à propager le RREQ comme spécifié dans AODV (voir figure 42).

Figure 42 - Exemple de requête QoS delay

Un nœud propageant le RREQ enregistre également l'adresse IP de la source dans la liste des sources demandant des garanties de délai ainsi que la valeur du délai demandé pour cette source. Une nouvelle comparaison pourra être ainsi effectuée en cas d'augmentation du NODE_TRAVERSAL_TIME. Cette nouvelle mesure sera initiée par un paquet spécifique RREP de perte de QoS (voir plus loin).

En réponse à une requête QoS RREQ, la destination envoie un paquet RREP avec un délai maximum initial nul (ainsi qu'une extension « Timestamp » pour la mesure du délai). Chaque nœud intermédiaire ajoute son propre NTT au champs Delay et enregistre cette valeur dans la table de routage pour la destination concernée avant de propager le RREP. Cette mise à jour d'entrée permet à un nœud intermédiaire de répondre à un RREQ suivant en comparant seulement la champs délai maximum de la table et la valeur de l'extension transmise (voir figure 43).

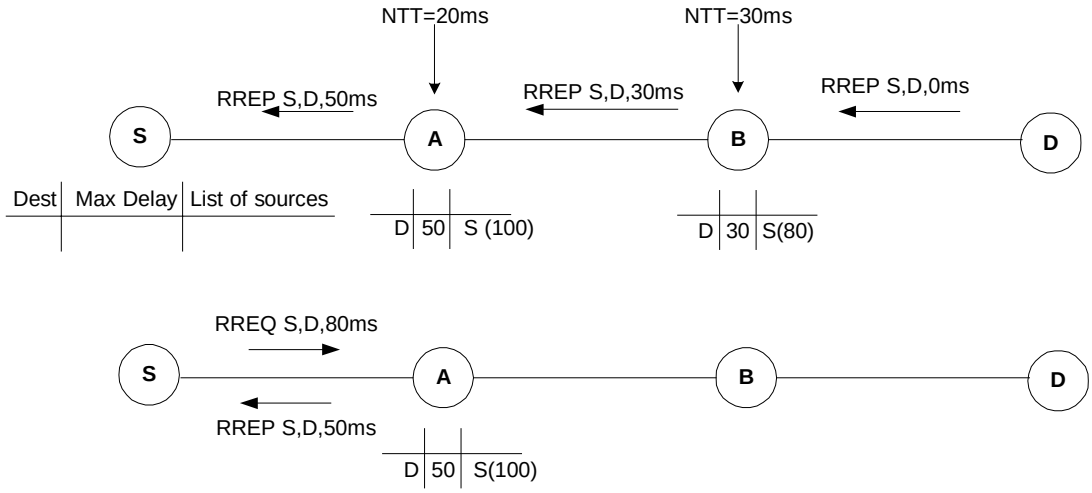


Figure 43 - Exemples de réponse QoS delay

Extension QoS Bande passante

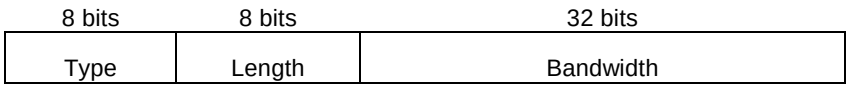


Figure 44 - Format de l'extension QoS Bande passante

Type : 7

Length : 4

Bandwith : deux significations pour ce champs suivant le type de paquet

- Pour un paquet RREQ, il indique la bande passante minimum (en kbit/s) qui doit être disponible sur tout le chemin entre la source et la destination
- Pour un paquet RREP, il indique la bande passante minimum disponible sur la route entre le nœud qui propage le RREP et la destination.

Un nœud avec des contraintes de bande passante transmet donc un paquet RREQ avec une extension QoS BP qui indique la bande passante minimum qui doit être disponible sur tout le chemin entre la source et la destination.

Avant de propager le paquet RREQ, un nœud intermédiaire compare sa capacité de lien disponible au champ BP indiqué dans l'extension. Si la BP demandée n'est pas disponible, le paquet est écarté et le processus s'interrompt

Si la BP demandée est disponible, le nœud continu à propager le RREQ comme spécifié dans AODV (voir figure 45).

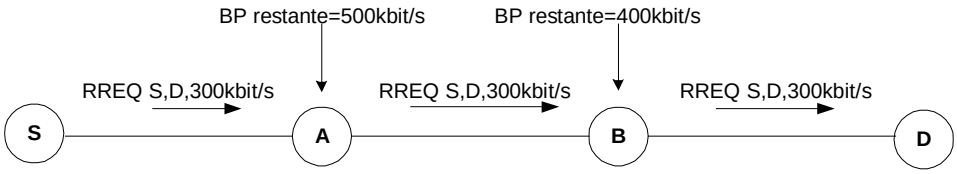


Figure 45 - Exemple de requête QoS BP

Comme pour le délai, les nœuds propageant les paquets RREQ enregistrent l'adresse IP de la source dans la liste des sources demandant des garanties de BP ainsi que la valeur de la BP demandée pour cette source. Une nouvelle comparaison initiée par un paquet spécifique RREP pourra être ainsi effectuée en cas de diminution de la BP restante.

En réponse à une requête QoS RREQ, la destination envoie un paquet RREP avec une BP minimum initiale infinie. Chaque nœud intermédiaire qui propage le RREP compare le champ BP de l'extension avec sa propre capacité de lien et garde le minimum entre ces deux valeurs pour propager le RREP. Cette valeur est également enregistrée dans la table de routage pour la destination concernée, elle indique la BP minimum disponible pour la destination. Cette mise à jour d'entrée permet à un nœud intermédiaire de répondre à un RREQ suivant en comparant seulement la BP minimum demandée dans l'extension QoS et le champ BP disponible enregistré dans la table (voir figure 46).

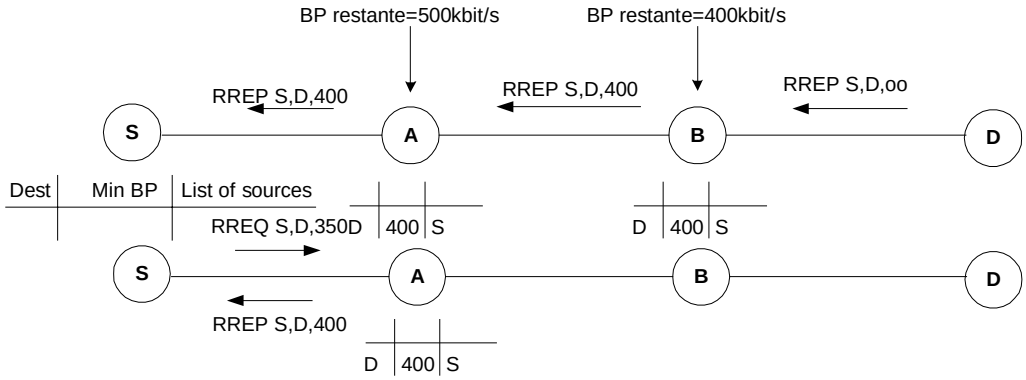


Figure 46 - Exemples de réponse QoS BP

Perte deQoS

Un paquet spécifique RREP muni d'une extension «Delay_increase » ou « Bandwith_decrease » est généré quand un nœud intermédiaire détecte une augmentation de son NTT ou une diminution de sa BP disponible qui ne permettent plus de garantir la QoS initialement demandée par une source. Dans la mesure où les nœuds ont enregistré dans la table de routage l'adresse des sources demandant des garanties ainsi que la valeur du délai ou de la BP demandé, le paquet RREP sera transmis à toutes les sources mémorisées et susceptibles d'être affectées par un changement de délai ou de BP détecté sur le nœud.

Dans l'exemple suivant, la valeur du NTT au nœud B passe de 30 à 90 ms. Le délai maximum dans la table passe ainsi à 90 ms et le délai demandé de 80 ms du nœud B à la destination (correspondant à la demande initiale de QoS de 100 ms de S vers D) ne peut plus être assuré, un paquet de perte de QoS doit être généré vers S.

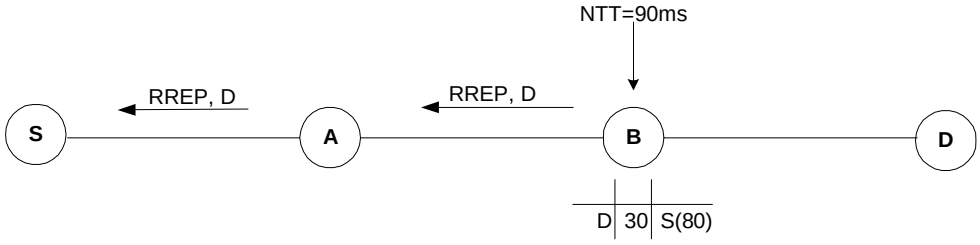


Figure 47 - Exemple de perte de QoS délai

5. Etude des performances de l'extension QoS sur AODV

5.1. Présentation de Network Simulator 2

Network Simulator 2 est un simulateur à événements discrets développé par le l'Université de Californie de Berkeley et le projet VINT. Il permet de simuler divers protocoles tels TCP sur des réseaux fixes conventionnels, mais aussi les aspects physiques et les protocoles du niveau MAC des réseaux mobiles sans fils. Quatre protocoles de routage pour les réseaux ad hoc sont implémentés dans ns2 : DSDV, DSR, TORA et AODV.

5.1.1. Modèles de la couche physique et de la couche liaison

Le modèle de propagation implémenté dans ns2 inclut un modèle de propagation en espace libre, dans lequel la puissance des signaux est atténuée en $\frac{1}{r^2}$ (r est la distance entre les antennes d'émission et de réception), et un modèle d'atténuation avec réflexion dans lequel la puissance est atténuée en $\frac{1}{r^4}$. Le premier modèle est appliqué lorsque les antennes sont éloignées d'une distance inférieure à la distance de référence qui est environ de 100m pour des antennes de faible gain placées à 1.5m au-dessus du sol et travaillant avec des fréquences de 1 à 2GHz.

Chaque nœud mobile est affecté d'une position et d'une vitesse et se déplace sur une topographie en relief ou plate. La position d'un mobile peut être calculée comme une fonction du temps et est utilisée par le modèle de propagation radio pour calculer le délai de transmission d'un nœud à un autre et le niveau de la puissance des signaux reçus par un nœud.

Chaque nœud possède une ou plusieurs interfaces réseau sans fil (NetIF), toutes les interfaces du même type de tous les nœuds étant reliées entre elles par un seul canal physique (*Channel*). Quand une interface réseau émet un paquet, elle le passe à un objet canal physique approprié. Cet objet calcule le délai de propagation depuis l'émetteur jusqu'à toutes les autres interfaces sur le canal et produit un événement « réception de paquet» pour chacune. Cet événement indique à l'interface de réception que le premier bit d'un paquet est arrivé. Le niveau de puissance du paquet reçu est comparé à deux valeurs différentes : le seuil de détection ou *carrier sense threshold* et le seuil de réception ou *receive threshold*. Si le niveau de puissance est inférieur au seuil de détection, le paquet est considéré comme du bruit et est écarté. Si le niveau de puissance est compris entre les deux seuils, il est marqué comme étant un paquet reçu avec erreur et est passé à la couche MAC. Sinon, le paquet est passé simplement à la couche MAC.

Quand la couche MAC reçoit le paquet, elle regarde si son état actuel est « au repos » ou *idle*. Si ce n'est pas le cas, deux choix sont possibles. Si le niveau de puissance du paquet reçu est inférieur de 10 dB au niveau de puissance du paquet actuellement traité, la couche MAC écarte le nouveau paquet et continue le traitement en cours. Sinon, elle considère qu'il y a collision et les deux paquets sont détruits.

Quand la couche MAC est au repos lors de l'arrivée d'un nouveau paquet, elle calcule le temps de transmission du paquet et génère pour elle-même un événement du type réception de paquet effectuée ou *packet reception complete*. Quand cet événement se produit, la couche MAC vérifie que le paquet est sans erreur, réalise le filtrage d'adresse de destination et remonte le paquet à la couche du protocole.

5.1.2. Méthode d'accès au support (MAC)

La couche liaison implémente la *Distributed Coordination Function* (DCF) de la norme IEEE 802.11 pour modéliser l'accés des nœuds au médium sans fil. DCF utilise à la fois les techniques dites *physical carrier sense* et *virtual carrier sense* pour réduire la probabilité de collision due au problème du nœud caché (*hidden node problem*). L'émission d'un paquet d'*unicast* est précédée d'un échange de RTS/CTS (*Request-To-Send/Clear-To-Send*) qui réserve la canal pour la transmission d'un paquet de données. Chaque paquet d'*unicast* reçu correctement est suivi d'un acquittement ACK émis vers l'émetteur qui retransmet le paquet un nombre limité de fois jusqu'à réception de l'acquittement. Les paquets de diffusion ne sont émis que lorsque les dites *physical carrier sense* et *virtual carrier sense* indiquent que le médium est libre, mqis ne sont pas précédés de RTS/CTS et ne font l'objet d'aucun acquittement.

5.1.3. Interface Queue (IFq)

Chaque nœud a une file de priorité du type FIFO contenant les paquets attendant une émission par la couche réseau, qui peut contenir 64 paquets et qui subit une gestion du type *drop-tail*. La file donne la priorité aux paquets de routage en les insérant en tête de file. Les protocoles réactifs TORA, DSR et AODV peuvent bufferiser séparément 50 paquets qui sont en attente d'une découverte de route à travers le réseau.

5.2. Implémentation de l'extension QoS sur AODV

Dans un premier temps seule l'extension « délai » sera implantée et les paquets de perte de QoS « delay increase » ne seront pas gérés.

Les fichier suivant sont modifiés :

- aodv.cc et aodv.h pour la gestion des extensions et de la QoS ;
- aodv_packet.h pour la définition des extensions sur les paquets RREQ et RREP ;
- rttable.h et rttable.cc pour l'ajout des entrées dans la table de routage ;
- cmu_trace.cc pour l'affichage du délai dans les fichiers de résultat.

Le calcul du délai est référencé au niveau réseau :

$d_{AB} = T_{recv} - T_{send}$

T_{send} correspond à l'instant d'émission du paquet (indiqué à B par une étiquette « Timestamp » dans l'extension) et T_{recv} à l'instant de réception.

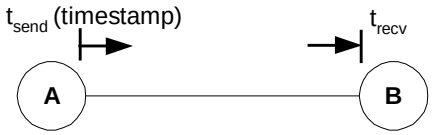


Figure 48 - Calcul du délai dans NS

L'algorithme du calcul de d_{AB} dans le nœud B devient :

A réception d'un paquet RREQ ou RREP (avec ou sans QoS) :

Lecture Timestamp dans extension

Calcul $d_{AB} = T_{recv} - Timestamp$

Correction longueur

Correction variance

Calcul $NTT_B = d_{AB} + t_{TB}$

Stockage NTT_B

La correction de longueur à 2Mbps avec une moyenne de 100 octets pour les paquets de données et de 24 octets pour les paquets RREQ ou RREP donne :

$d'_{AB} = d_{AB} + LENGTH\_COR$

avec $LENGTH\_COR = (100-24)*8 / 2.10^6 = 0.304$ ms

La correction de variance est appliquée à l'ordre 1 :

$$D_{AB}(t) = (1-\lambda) d_{AB}(t) + (1-\lambda)\lambda D_{AB}(t-1)$$

Après une série de mesures (voir résultats), les différents paramètres sont initialisés avec les valeurs suivantes :

- FORGET_FACTOR (facteur d'oubli λ) = 0,2
- PROCESS_TIME (temps de traitement dans le nœud) = 3 ms

Le temps de traitement dans le nœud est mesuré dans la simulation entre l'instant de réception et l'instant d'émission d'un paquet RREQ. Les temps de traitement sont nuls dans la simulation pour les « forward » de paquets de données ou de réponse RREP.

5.3. Résultats

Les mesures sont effectuées avec un nombre de nœuds compris entre 10 et 50 repartis au hasard sur espace de 670 m x 670 m.

Chaque nœud est susceptible d'être à un instant donné une source CBR à 4 paquets de 512 octets par seconde.

Les liens sont à 2 Mbit/s et les portées de 250 m.

Les déplacements se font aléatoirement toutes les 30 secondes à des vitesses comprises entre 1 et 10 m/s.

Les requêtes de route avec QoS sont réalisée avec des demandes de délai variant de 10 à 100 ms.

La durée des simulations est de 300 s.

5.3.1. Evolution du NTT

La première série de mesures donne l'évolution du NODE_TRAVERSAL_TIME pour un nœud donné sur une fenêtre de temps de 40 ms correspondant à 10 échantillons successifs. Le nombre de nœuds est de 20 pour 7 sources CBR (environ 1/3). Les nœuds se déplacent à une vitesse maximum de 10 m/s.

La première courbe montre l'évolution du NTT sans correction (facteur d'oubli nul) ainsi que la moyenne sans pondération (moyenne simple de la mesure courante avec la précédente). La deuxième courbe montre l'évolution du NTT pour différentes valeurs du facteur d'oubli.

Nous constatons que pour un facteur d'oubli supérieur ou égal à 0,3, les valeurs du NTT baissent fortement et l'algorithme appliqué à l'ordre 1 n'est plus adapté à l'évolution du délai. Nous retiendrons un facteur d'oubli de 0,2 qui offre le meilleur compromis pour la prise en compte des variations dans le temps du délai de nœud à nœud.

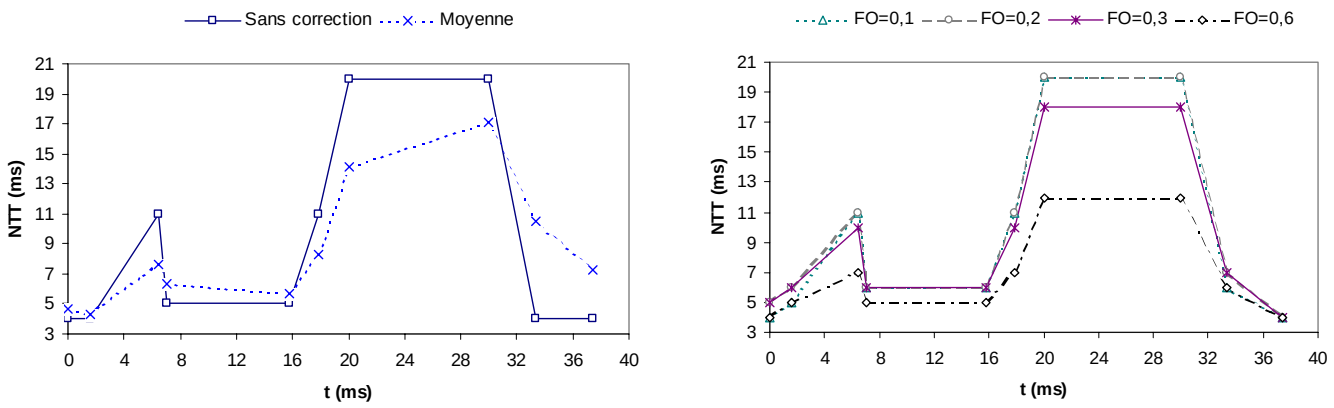


Figure 49 - Evolution du NTT en fonction du facteur d'oubli

5.3.2. Variation du délai QoS

La deuxième série de mesures donne en pourcentage le nombre de paquets routés et le nombre moyen de sauts nécessaires en fonction de la valeur du délai QoS demandé par les sources. Le nombre de nœuds et de 10 ou 50 pour environ 1/3 de sources (3/10 ou 17/50). Chaque nœud est susceptible de se déplacer aléatoirement à une vitesse maximum de10 m/s.

Sans demande de QoS, le nombre de paquets routés est d'environ 72 % pour 10 nœuds et 99% pour 50 nœuds. Ces résultats s'expliquent par le nombre relativement faible de sources (au delà d'1/3 le taux de paquets routés chute notablement, voir figure 55) et par une plus grande densité dans le même espace pour le réseau de 50 nœuds.

Avec demande de délai QoS, le taux de succès augmente presque linéairement dans un réseau de 50 nœuds lorsque la contrainte QoS diminue ; pour 10 nœuds, le taux reste faible (inférieur à 65 %) lorsque le délai requis est inférieur à 50 ms et chute à près de 50 % pour une contrainte inférieure à 20 ms. Les taux atteints pour un délai de 100 ms sont proches pour les deux densités de ceux obtenus sans QoS.

Les courbes donnant le nombre moyen de sauts en fonction du délai QoS confirment ces résultats. Avec un NTT moyen d'environ 8 ms, le point critique de routage QoS se situe vers 20ms pour un nombre moyen de 2 sauts.

Par ailleurs, compte tenu de la densité relativement importante de nœuds (10 ou 50 sur une surface de 670 x 670 m) et de la portée de chacun (250 m), le nombre moyen de sauts pour transmettre un paquet de bout en bout reste globalement faible (entre 1,25 et 2,22).

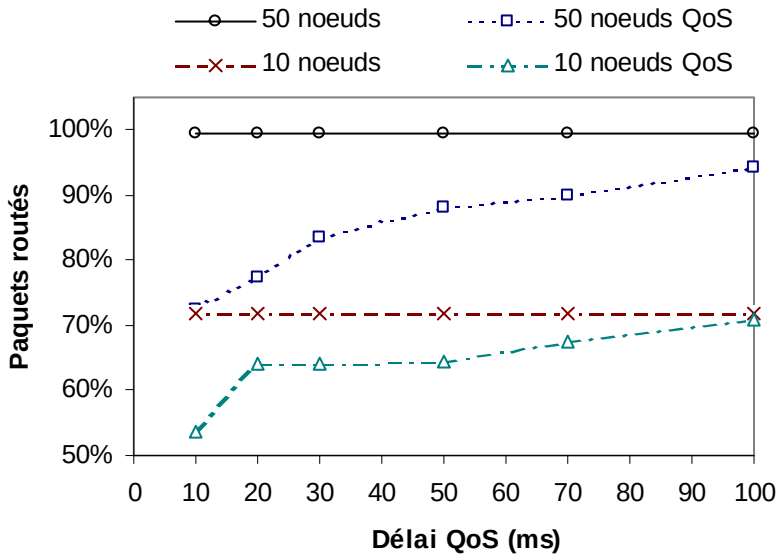


Figure 50 - Paquets routés en fonction du délai QoS

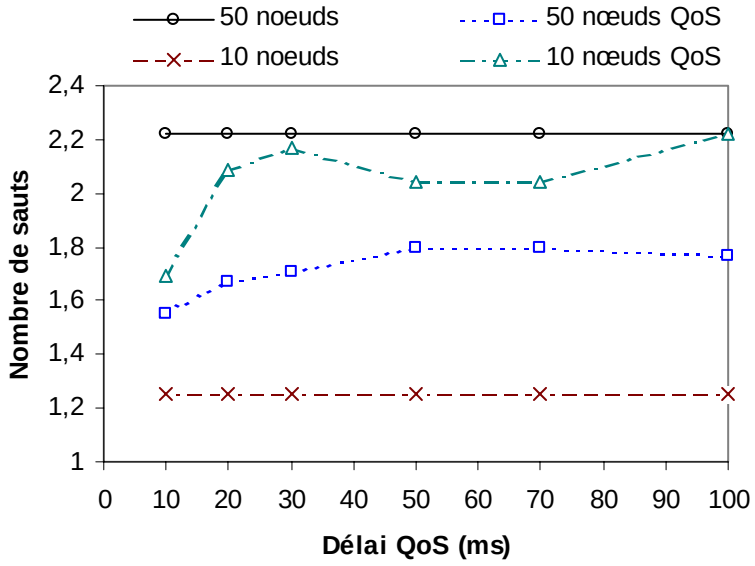


Figure 51 - Nombre moyen de sauts en fonction du délai QoS

5.3.3. Variation du nombre de sources

Les courbes suivantes (figures 52 et 53) représentent l'évolution du délai moyen de bout en bout pour la transmission des paquets de 512 octets et du débit atteint en fonction du nombre de sources exprimé en pourcentage du nombre total de nœuds dans le réseau (10 ou 50). La vitesse maximum est de 10 m/s et les délais QoS demandés sont de 100 ms.

Le délai moyen augmente faiblement (7 à 10 ms) pour un réseau de 10 nœuds avec ou sans QoS tant que nombre relatif de sources est inférieur à 40% ; les deux mesures suivantes (6 et 8 sources pour 10 nœuds) correspondent à des scénarios de simulation donnant une saturation relative du réseau (délais de 200 à 300 ms).

Pour un réseau de 50 nœuds, le délai moyen augmente également peu (7 à 14 ms), le nombre croissant de sources étant compensé par la densité du réseau et leur répartition (voir figure 54), et la saturation n'est pas atteinte dans ces conditions de vitesse et débit des sources (voir variation de la vitesse figure 57).

Dans tous les cas, les valeurs obtenues avec QoS ne dépassent que légèrement celle obtenues sans QoS.

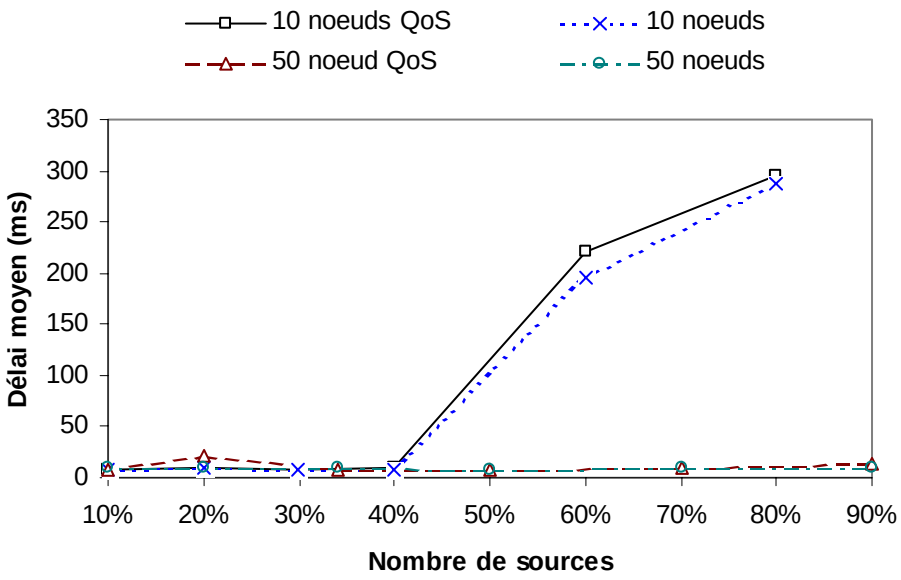


Figure 52 - Délai moyen de bout en bout en fonction du nombre de sources

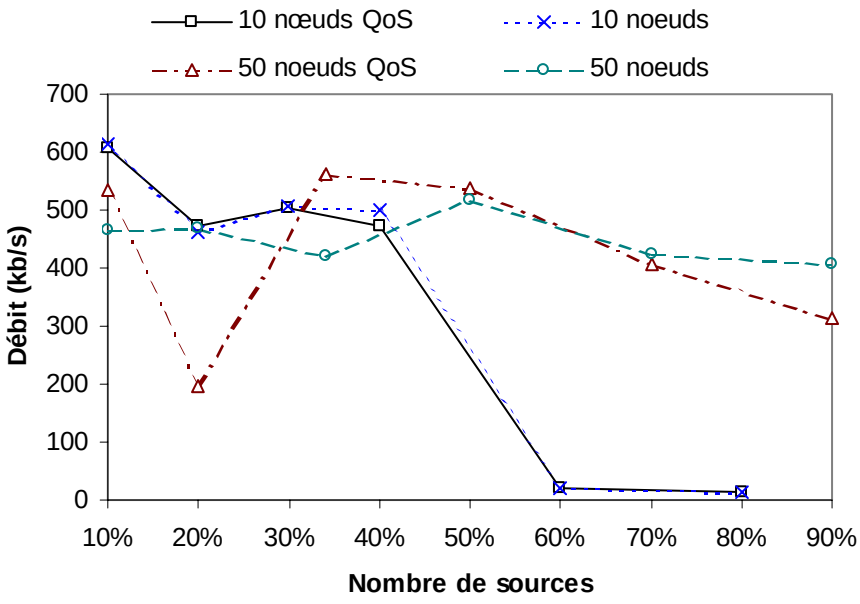


Figure 53 - Débit moyen en fonction du nombre de sources

La figure 54 donne le nombre moyen de sauts lors de la transmission d'un paquet de bout en bout en fonction du nombre relatif de sources. La densité du réseau de 50 nœuds explique à nouveau un nombre légèrement inférieur (proche de 2) par rapport au réseau de 10 nœuds (moyenne de 2,3). Dans la mesure où le taux de paquets routés est moins important avec QoS (voir figure 55) le nombre moyen de sauts est légèrement inférieur pour 50 nœuds.

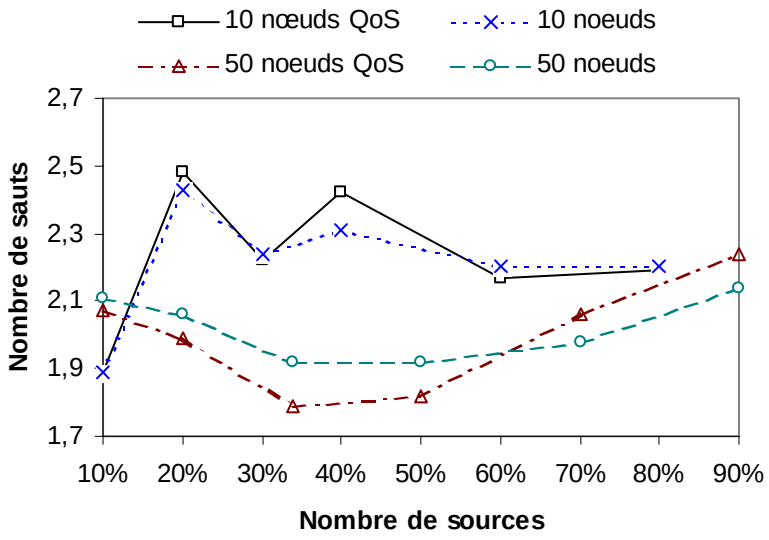


Figure 54 - Nombre moyen de sauts en fonction du nombre de sources

La figure 55 représente le taux de paquets routés. Sans demande de QoS, environ 99% des paquets émis sont reçus, quelque soit le nombre relatif de sources, sur un réseau de 50 nœuds. Ce taux reste relativement constant autour de 95% avec QoS ce qui confirme l'absence de saturation du réseau dans ces conditions.

Pour un réseau peu dense de 10 nœuds, les taux sont inférieurs notamment pour un faible nombre de sources dans la mesure où un nœud peut être facilement isolé des autres (portée de 250 m).

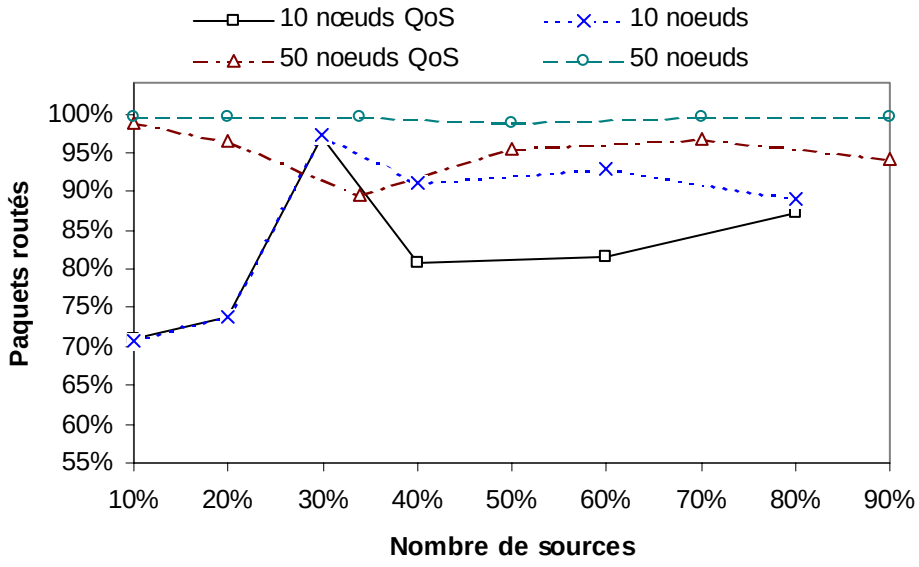


Figure 55 - Paquets routés en fonction du nombre de sources

5.3.4. Variation de la vitesse

Les courbes suivantes (figures 56 et 57) représentent l'évolution du délai moyen de bout en bout pour la transmission des paquets de 512 octets et du débit atteint en fonction de la vitesse de déplacement des nœuds ; les valeurs données sont des valeurs maximales avec une variation de 3m/s (une vitesse de 5 m/s correspond pour chaque nœud à une vitesse tirée aléatoirement et comprise entre 2 et 5 m/s). Les déplacements se font toutes les 30 s pour une durée de mesure de 300 s. Le nombre relatif de sources est fixé à 1/3 (3 pour 10 nœuds et 17 pour 50 nœuds) et les délais QoS demandés sont de 100 ms.

Les valeurs de délai restent proches de 10 ms pour une densité de 10 nœuds quelque soit la vitesse. Pour 50 nœuds, une saturation est observée à partir de 3 m/s dans le cas de requêtes QoS et de 7 m/s sans QoS et les délais augmentent fortement (200 ms). Cette saturation n'apparaît pas dans la figure 52 qui correspond pourtant pour un des points aux mêmes conditions de mesures (1/3 de sources et 10 m/s) ; dans ce cas, les vitesses sont uniformément réparties entre 0 et 10 m/s et non entre 7 et 10 m/s.

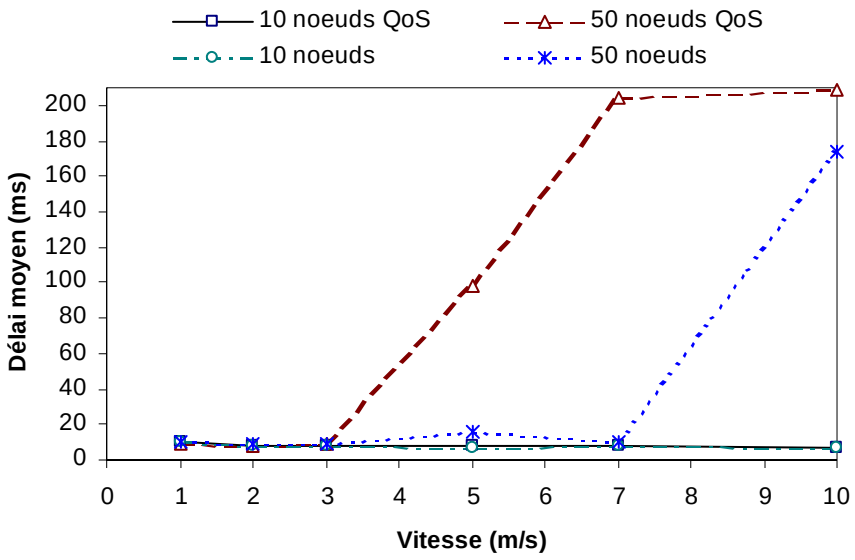


Figure 56 - Délai moyen de bout en bout en fonction de la vitesse

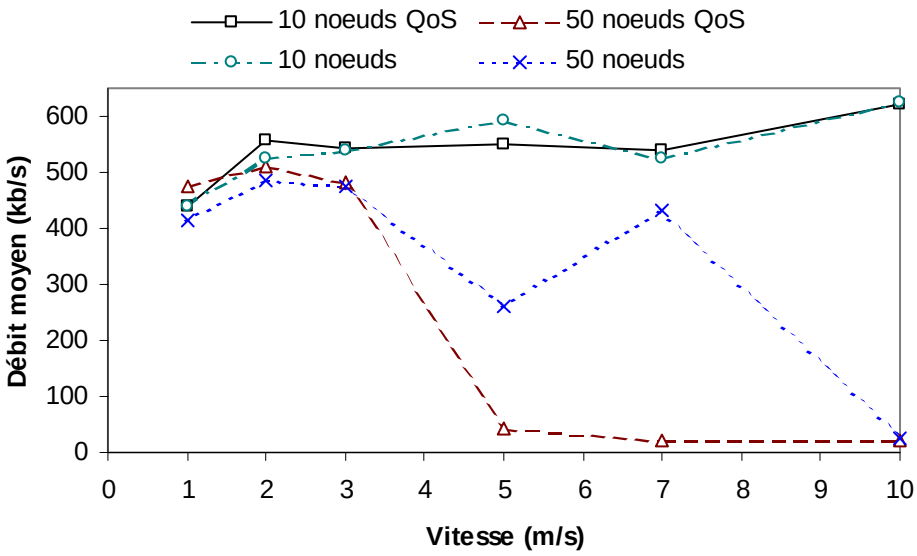


Figure 57 - Débit moyen en fonction du nombre de la vitesse

La figure 58 donne le nombre moyen de sauts lors de la transmission d'un paquet de bout en bout en fonction de la vitesse. Les valeurs obtenues sont toujours légèrement plus faibles pour un réseau plus dense (moyenne de 2,2). Pour de faibles vitesses, le nombre de sauts est plus important (environ 3 pour 10 nœuds) ce qui correspond à une situation de départ moins favorable avant que sources et destinations se rapprochent.

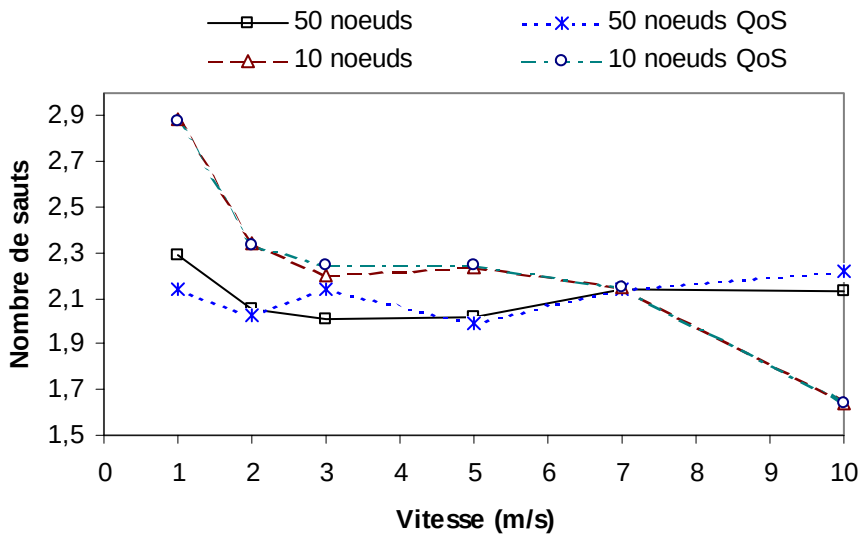


Figure 58 - Nombre moyen de sauts en fonction de la vitesse

La figure 59 donne le taux de paquets routés. Sans demande de QoS, environ 99% des paquets émis sont reçus, quelque soit la vitesse, sur un réseau de 50 nœuds. Ce taux reste supérieur à 90 % avec des sources QoS ce qui semble cohérent avec des requêtes de délai de 100 ms.

Pour un réseau peu dense de 10 nœuds, les taux chutent avec ou sans QoS à partir de 5 m/s et les routes ne sont pas trouvées pour toutes les sources.

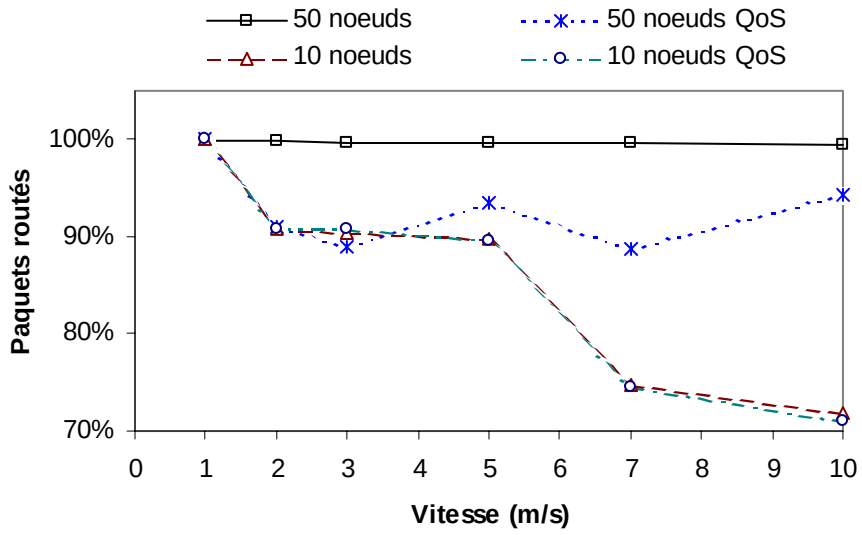


Figure 59 - Paquets routés en fonction de la vitesse

5.3.5. Conclusion sur les performances

Ces premiers résultats confirment la viabilité de cette solution et montrent que l'ajout d'une extension QoS sur les paquets de requête et de réponse affecte relativement peu les performances. Les taux de paquets routés restent satisfaisants (supérieurs à 90%) dans la plupart des conditions de mesure.

Pour compléter ou poursuivre ces mesures, plusieurs axes peuvent être envisagés :

- réalisation de scénarios multiples de variation du nombre de sources et de la vitesse afin d'effectuer des moyennes pour éviter les points aberrants ;

- augmentation des vitesses, des débit des sources, des contraintes QoS et de la surface du réseau pour atteindre la saturation ou provoquer une plus grande perte de paquets ;
- mesures avec variation du temps de pause avant déplacement des nœuds ;
- mesures comparatives avec d'autres protocoles QoS

Par ailleurs, la totalité des procédures QoS devront être implantées :

- Extension QoS bande passante
- Gestion des paquets de perte de QoS

Parmi les améliorations possible mises en évidence :

- Tous les paquets ont RREP et RREQ ont une extension QoS, y compris ceux pour lesquels aucune requête QoS n'est demandée par la source. Une solution consiste à rajouter un bit dans le champs réservé de l'en-tête des paquets indiquant si l'extension est présente ou non.
- Prise en compte des variations du temps de traitement dans les nœuds (celui-ci est pour l'instant considéré comme une constante). Ce point est à relié aux limites du simulateur pour lequel les temps de traitement sont considérés comme nuls pour les « forward » de paquets de réponse RREP ou de données.

6. Conclusion

Nous avons constaté au cours de cette étude les insuffisances des protocoles de routage classiques pour les réseaux ad hoc, qui ne tiennent nullement compte de l'état du réseau et des exigences des applications lors de la recherche et de la maintenance des routes. De nouvelles approches, très diverses, incorporant des critères de qualité de service ont été proposées. Cependant, elles présentent pour la plupart l'inconvénient de ne traiter qu'un seul type de contrainte à la fois, et surtout de ne pas être adaptées aux couches inférieures.

Notre travail a consisté à mettre au point un mécanisme qui prend en compte plusieurs critères de qualité de service à la fois, et qui s 'appuie sur une couche MAC 802.11 fortement implantée à l'heure actuelle. Cette nouvelle approche, a été superposée au protocole réactif AODV. Les premiers résultats de simulation obtenus sont encourageants et les améliorations mises en évidence devraient augmenter les performances.

D'autres propositions telles SWAN [39] et INSIGNIA [40] visant à intégrer la qualité de service dans un contexte plus global sont également à étudier.

7. Annexes

7.1. La norme IEEE 802.11

7.1.1. Architecture

Le protocole 802.11 définit 2 catégories d'équipements :

- Les stations sans fils (*wireless station*) : PC, portable ou PDA équipées d'une WNIC (*Wireless Network Interface Card*).
- Les points d'accès (AP - *Acces Point*) qui coordonnent les transmissions et servent de pont entre le réseau câblé et le WLAN.

Il est basé sur une topologie cellulaire (le système est subdivisé en cellules) où chaque cellule (BSS - *Basic Service Set*) est contrôlée par un AP.

Le protocole 802.11 implémente 2 modes de fonctionnement.

- **Le mode infrastucture** (*Infrastructure Mode*). Dans la plupart des installations, le WLAN est composé de plusieurs cellules où chaque AP de cellule est interconnecté aux autres par une dorsale (backbone) câblée (DS - *Distribution System*). Cette interconnexion correspond à un ensemble étendu de cellules (ESS - *Extended Service Set*).

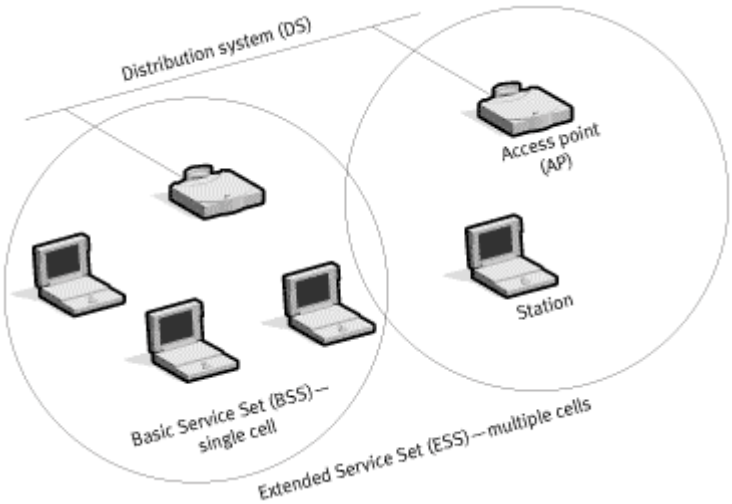


Figure 60 - Mode infrastructure

- **Le mode ad hoc**, également appelé mode sans infrastructure ou IBSS (*Independent Basic Service Set*), permet à des stations de communiquer directement entre elles sans utiliser un point d'accès. Ce mode simplifié permet de réaliser rapidement une communication entre 2 stations sans fils. Pour pouvoir fonctionner sur un réseau étendu, ce mode doit être associé à un protocole de routage permettant à une station de communiquer avec une station éloignée par l'intermédiaire de stations faisant office de routeur.

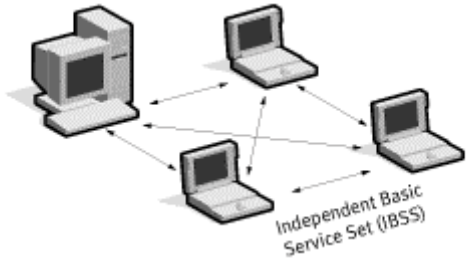


Figure 61 - Mode ad hoc

Association à une cellule existante

Quand une station veut accéder à un BSS ou à un IBSS, soit après démarrage, un passage en mode veille ou un déplacement, la station à besoin d'informations de synchronisation de la part du point d'accès (ou des autres stations dans le cas du mode ad hoc). La station peut obtenir ces informations suivant deux méthodes choisies en fonction de la puissance des signaux reçus ou de la consommation d'énergie engendrés par l'échange :

- Ecoute passive : la station attend de recevoir une trame balise (*Beacon Frame*) envoyée périodiquement par l'AP et contenant les informations de synchronisation.
- Ecoute active : la station essaie de rejoindre un AP en transmettant une trame de requête (*Probe Request Frame*) et attend la réponse du point d'accès.

La phase d'association se poursuit par un processus d'authentification qui peut être sécurisé par un échange de trames cryptées. Le processus se termine par un échange d'informations concernant les caractéristiques de la cellule et la station peut transmettre et recevoir des trames de données. L'émission périodique de trames balise assure la synchronisation tout au long des échanges.

Le ***handover*** est le processus de déplacement d'une station d'une cellule vers une autre sans interruption de la communication. Ce processus n'est pas directement géré par 802.11 mais les mécanismes d'association permettent de supporter cette mobilité. La station teste la puissance du signal de l'AP et si celui-ci s'affaiblit, elle se ré-associe avec un nouvel AP sur un autre canal. Ce processus dynamique d'association et de ré-association avec les AP permet de mettre en place des WLAN de couverture importante en créant une série de cellules se recouvrant. Il faut alors éviter de configurer 2 AP voisins avec le même canal pour éviter les interférences. Lors d'un déplacement en mode ad hoc, les stations testent de la même façon la présence d'autres stations sur différents canaux.

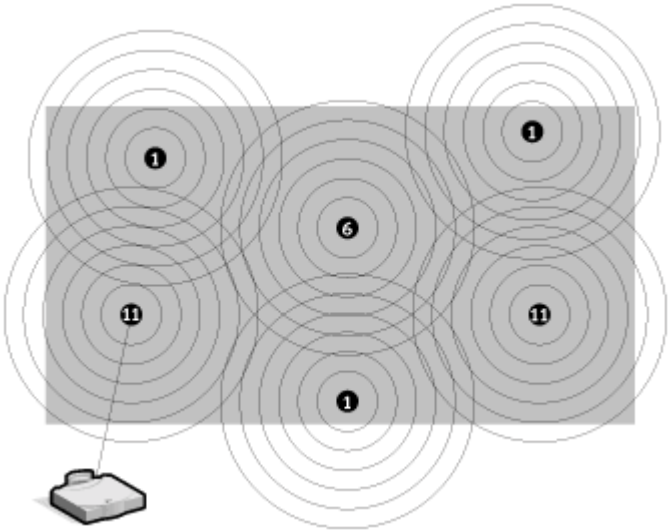


Figure 62 - Affectation des canaux aux BSS ou IBSS

7.1.2. Modèle IEEE 802.11

La norme IEEE 802.11 concerne les couches PHY et MAC du modèle OSI.

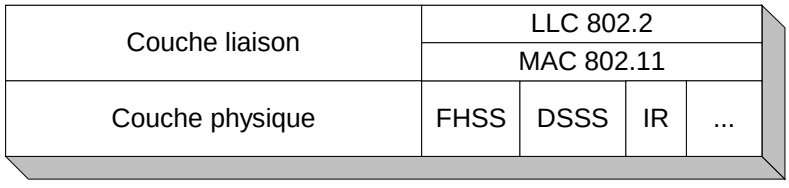


Figure 63 - Modèle IEEE 802.11

7.1.3. La couche physique

Le standard définit actuellement une seule couche MAC qui interagit avec 3 couches physiques.

- **IR** (infrarouge).

- **FHSS** (*Frequency Hopping Spread Spectrum*). La bande sans licence ISM (Industrie, Science et Médecine) des 2,4 GHz est divisée en 79 canaux de 1 MHz chacun (limité à 35 canaux en Frânçe). La transmission se fait sur toute la largeur de la bande avec un saut de fréquence (un changement de canal) toutes les 20 ms suivant une séquence commune au sein d'une même BSS. Cette technique accroît l'immunité au bruit et permet une co-localisation théorique de 15 BSS (1 canal par BSS). Le débit est limité à 2 Mbit/s.

- **DSSS** (*Direct Sequence Spread Spectrum*). La technique de la séquence directe divise la bande des 2.4 GHz en 14 canaux de 20 MHz chacun (limité à 4 canaux en Frânçe). La transmission ne se fait que sur un seul canal par BSS. Pour compenser le bruit, une technique de *chipping* qui consiste à convertir chaque bit de données en une séquence de 11 bits est utilisée. Cette technique permet une co-localisation théorique de 3 BSS.

Les modulations à saut de phase appliquées aux séquences permettent d'obtenir des débits différents. La BPSK (*Binary Phase Shift Keying*) pour un débit de 1 Mbit/s et la QPSK (*Quadature Phase Shift Keying*) pour un débit de 2 Mbit/s.

Dans le protocole 802.11b, pour pouvoir supporter les débits de 5.5 Mbit/s et 11 Mbit/s, la technique DSSS High-rate est utilisée. Cette augmentation des débits est réalisée en ajoutant à une modulation QPSK une technique de codage CCK (*Complementary Code Keying*).

Pour une réception satisfaisante, la portée de l'émetteur ne doit pas dépasser 150 m dans un environnement de bureau et 600 m sans obstacles. En pratique, on limite les distances à 50 m pour garder une qualité de réception optimale.

7.1.4. La couche MAC

La sous-couche MAC est unique au protocole 802.11. Elle définit 2 fonctions de coordination des échanges correspondant à 2 méthodes d'accès différentes :

- **PCF** (*Point Coordination Function)* est basé sur l'interrogation à tour de rôle des terminaux (*polling*) par l'AP. Ce mode utilisé en alternance avec un mode distribué (DCF) est conçu pour des applications de type temps réel, telles que la voix ou la vidéo.

- **DCF** (*Distributed Coordination Function)* n'est pas fondé sur une gestion centralisée et permet la prise en charge du transport de données asynchrones avec des chances égales pour toutes les stations d'accéder au support (type *best effort*). Un réseau ad hoc utilise uniquement le DCF.

Les liaisons radio n'étant pas full duplex, une méthode de détection de collision de type CSMA/CD ne peut être utilisée dans la mesure ou une station ne peut être à l'écoute du support pendant son émission.

Un mécanisme d'écoute de porteuse avec évitement de collision et acquittement est donc utilisé. DCF correspond à la méthode d'accès **CSMA/CA** (C*arrier Sense Multiple Acces with Collision Avoidance*).

Une station voulant transmettre écoute le support, s'il est occupé, la transmission est différée. Si le support est libre pendant un temps supérieur au **DIFS** (*DCF Inter Frame Spacing*), la station est autorisée à transmettre. La station réceptrice va vérifier le CRC du paquet reçu et renvoie un accusé de réception **ACK**. La réception de l'ACK indiquera à l'émetteur qu'aucune collision n'a eu lieu. Si l'émetteur ne reçoit pas l'accusé de réception, alors il retransmet la trame jusqu'à ce qu'il l'obtienne ou abandonne au bout d'un certain nombre de retransmissions.

Pour séparer les transmissions au sein d'un même dialogue (données, ACK…) un temps inter-trame plus faible **SIFS** (*Short IFS*) est suffisant dans la mesure où seule une station est susceptible d'émettre à cet instant (émetteur ou récepteur en cours). Un troisième temps inter-trame **PIFS** (*PCF IFS*) inférieur au DIFS peut être utilisé par l'AP pour accéder prioritairement au support.

Pour éviter une collision, les stations qui entendent une transmission en cours utilisent un temporisateur régulièrement mis à jour, le **NAV** (*Network Allocation Vector)* calculé en fonction du champs délai contenu dans les trames émises. Les stations voulant émettre et trouvant le support encore occupé après le temps DIFS attendent donc un temps correspondant au NAV (durée théorique de l'occupation) augmenté d'un nouveau DIFS (voir figure 64). Au bout de ce temps d'attente, les stations ne cherchent pas à émettre toutes en même temps et évitent les collisions en attendant chacune un temps supplémentaire aléatoire suivant un algorithme de *backoff*.

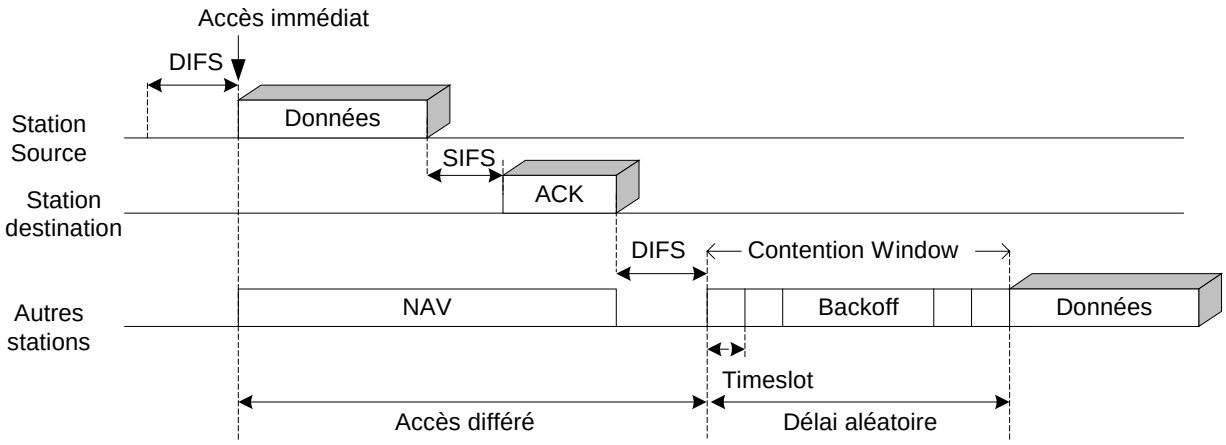


Figure 64 - Accès au support CSMA/CA

L'algorithme de *backoff* est basé sur la gestion de *timeslots* correspondant à des tranches de temps fixées par la couche physique. Initialement, une station calcule en nombre de *timeslots* la valeur d'un temporisateur (*timer backoff*) compris entre 0 et CW_{max} (*Contention Window*). Lorsque le support est libre pendant un temps supérieur au DIFS, la station décrémente son temporisateur jusqu'à ce que le support soit occupé par une autre station avec un *timer* plus faible (la décrémentation est suspendue) ou que son propre temporisateur expire (c'est alors elle qui émet).

Si deux stations tirent le même nombre de *timeslots*, la collision ne peut être évitée, elle est alors détectée par l'absence de réponse ACK. Le *backoff* est dit exponentiel : à chaque fois qu'une station choisit un nombre de slots et provoque une collision, le nombre maximum pour la sélection aléatoire est augmenté exponentiellement.

Timer backoff = $[2^{2+i}$ x rand()] x timeslot

En définitive, l'algorithme de *backoff* exponentiel est exécuté dans les cas suivant :

- quand la station écoute le support avant la première transmission d'un paquet et que le support est occupé ;
- après chaque retransmission ;
- après une transmission réussie ;
- après une détection de collision.

Le seul cas où ce mécanisme n'est pas utilisé est quand la station décide de transmettre un nouveau paquet et que le support a été libre pour un temps supérieur au DIFS.

Problème de la station cachée

Une station A émet vers une station B ; une autre station C qui est hors de portée de la station A n'entends pas l'émission et risque de vouloir émettre à son tour vers la station B et donc de provoquer une collision qui ne peut être éviter par la méthode CSMA/CA (voir figure 65).

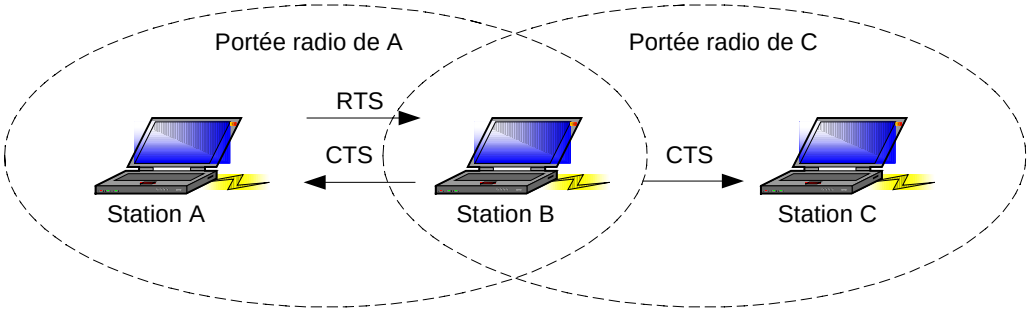


Figure 65 - Station cachée et détection virtuelle de porteuse par RTS/CTS

Pour résoudre ce problème, la norme 802.11 a défini le mécanisme de **VCS** (*Virtual Carrier Sense*). Il est basé sur l'un des premiers protocoles développés pour les WLANs : MACA (*Multiple Access with Collision Avoidance*). Ce mécanisme virtuel de détection de porteuse au niveau MAC est comparable à l'écoute du support effectué au niveau physique (PCF - *Physical Carrier Sense*).

Une station voulant émettre transmet d'abord une petite trame (30 octets) de contrôle RTS (*Request To Send*). Toutes les stations qui entendent le RTS mettent à jour leurs NAV en fonction du champs durée du RTS. La station destination concernée répond après attente d'un temps SIFS avec une trame courte CTS (*Clear To Send*). Le NAV est de nouveau mis à jour par les stations entendant le CTS. Après réception du CTS par la station source, celle-ci est assurée que le support est réservé pour sa transmission pendant un temps au moins égal au NAV. Par ailleurs, les stations cachées hors de portée de l'émetteur seront prévenues d'une émission en cours (dans l'exemple précédent, la station C qui aura entendu le CTS émis par B différera sa transmission). Ce mécanisme permet donc d'une part de réserver le support pendant un temps paramétrable et de résoudre le problème de la station cachée. Il n'évite cependant pas les collisions de RTS ou de CTS, mais celles-ci sont moins coûteuse que des collisions de longues trames de données.

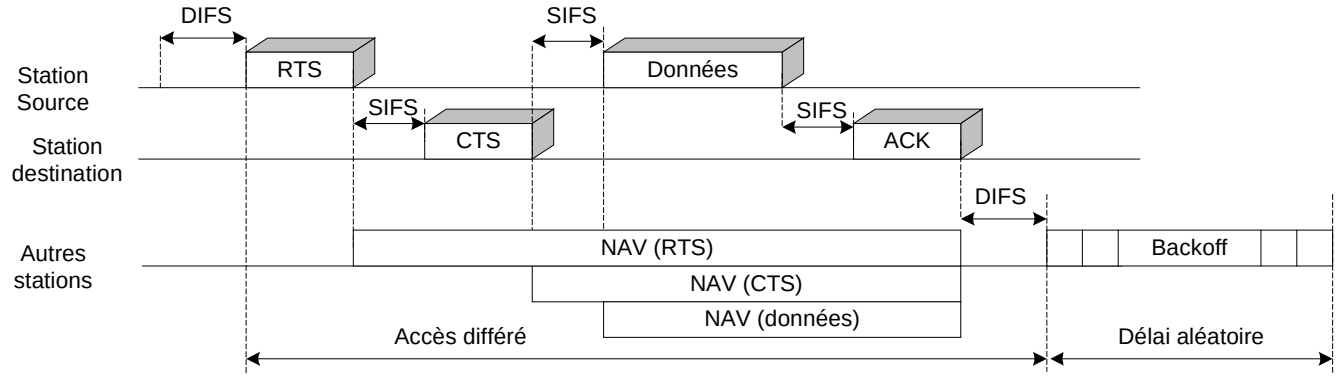


Figure 66 - Réservation de support avec les trames RTS/CTS

Fragmentation-réassemblage

Les protocoles de réseaux locaux filaires utilisent des trames de plusieurs centaines d'octets (1518 octets pour Ethernet). Dans un environnement de réseau local sans fil, il y a des plusieurs raisons d'utiliser des trames plus petites :

- le taux d'erreur par bit est plus important sur une liaison radio, la probabilité pour qu'une trame soit corrompue augmente avec sa taille ;
- dans le cas d'une trame corrompue (à cause d'une collision ou même du bruit), plus sa taille est faible, moins le surdébit engendré par sa retransmission est important ;
- dans un système à saut de fréquence, le support est interrompu toutes les 20 ms. Plus la trame est petite, plus la probabilité d'avoir une transmission interrompue est faible.

Ce mécanisme de fragmentation et réassemblage est introduit au niveau de la couche MAC lorsque la taille des trames dépasse un seuil (*Fragmentation Thresold*). Il se résume à un algorithme simple d'envoi et d'attente d'acquittement. La station émettrice garde le contrôle du support pendant la durée d'émission de tous ses fragments. Chaque fragment doit être acquitté par un ACK avec des temps inter-trame égaux à SIFS.

7.1.5. Sécurité

La norme 802.11 implémente au niveau MAC un processus de sécurisation optionnel WEP (*Wired Equivalent Privacy*) qui intervient à deux niveaux :

- Lors de la phase d'authentification, avec un échange de texte crypté à l'aide d'une clé secrète partagée (*Shared Key Authentication*).
- Pour empêcher une écoute clandestine lors de l'échange de données (contrairement à un réseau filaire, toute station non connectée située dans une zone de transmission peut tenter d'écouter). Le mécanisme de chiffrement est basé sur l'algorithme RC4 et utilise un générateur de nombres pseudo aléatoires initialisé par une clef secrète partagée et codée sur 64 bits. Le générateur ressort une séquence de clefs pseudo aléatoires qui permet de chiffrer les données de manière différente à chaque transmission.

7.1.6. Trames IEEE 802.11

Il y a trois principaux types de trames :

- les trames de données qui encapsulent les paquets IP ;
- les trames de contrôle, utilisées pour contrôler l'accès au support (RTS, CTS, ACK) ;
- les trames de gestion, utilisées pour les phases d'association et d'authentification.

La structure de base des trames est commune et comporte les différents champs :

- Le préambule est dépendant de la couche PHY et correspond à une séquence permettant de synchroniser et de définir le début de trame ;
- L'en-tête PLCP (*Physical Layer Convergence Protocol*) est utilisé par la couche PHY pour décoder la trame , il contient des information sur la longueur de la trame et le débit des données.

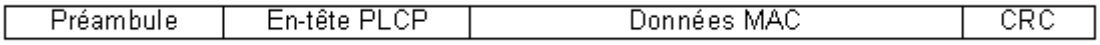


Figure 67 - Structure de base des trames 802.11

Les trames de données et de gestion comportent un certain nombre de champs dans l'en-tête MAC :

- Les octets de contrôle de trame donnent des indications telles la version du protocole, le type de trame, la direction de la trame (bit ToDS à 1 lorsque l'AP doit relayer la trame vers le système de distribution DS ; bit FromDS à 1 lorsque la trame vient d'une station), une fragmentation, une retransmission, un cryptage WEP…
- Le champs durée donne la valeur utilisée pour le calcul du NAV.
- Les champs Adresse ont différentes significations suivant les valeurs des bit ToDS et FromDS ; Adresse1 et Adresses 2 correspondent toujours aux adresses récepteur et émetteur.
- Le champ de contrôle de séquence est utilisé pour représenter l'ordre des différents fragments appartenant à la même trame et pour reconnaître les paquets dupliqués.

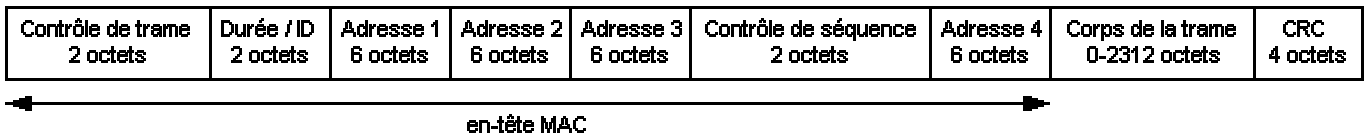


Figure 68 - Structure de la zone de données MAC

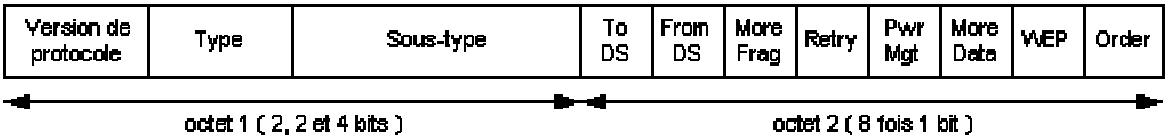


Figure 69 - Octets de contrôle de trame de l'en-tête MAC

Les trames de contrôle les plus courantes ont un format limitée

Pour la trame RTS, RA est l'adresse du récepteur de la prochaine trame de données ou de gestion. TA est l'adresse de l'émetteur. La valeur de la durée est le temps, en microsecondes, nécessaire à la transmission d'une trame CTS, d'une trame de gestion ou de données et d'une trame ACK augmenté de 3 intervalles SIFS (voir NAV-RTS figure 66).

La structure de trames CTS et ACK est similaire. Dans la mesure où ce sont des trames de réponses, les 6 octets TA sont absents.

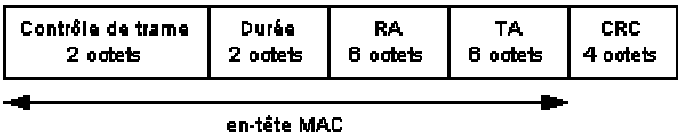


Figure 70 - Format de la trame RTS

7.1.7. Evolutions

Le tableau suivant résume les versions en cours et les évolution futures de la norme. La version commercialisée à l'heure actuelle est la 802.11b à 11 Mbit/s. Le mécanisme de réservation de support avec les trames RTS/CTS est rarement implanté sur les cartes et l'algorithme de chiffrement WEP est considéré comme peu robuste.

Les évolutions portent surtout sur les débits qui doivent être associés à d'autres type de codage ou modulations (*Othogonal Frequency Division Multiplexing* pour le 802.11a), la QoS (802.11e), une meilleure gestion du handover (802.11f) ou la sécurité (802.11i).

Version	**802.11**	**802.11b ou Wi-Fi**	**802.11g**	**802.11a ou Wi-Fi5**	**802.11e**	**802.11f**	**802.11i**
Débit	1-2 Mbit/s	2-11 Mbit/s	2-54 Mbit/s	6-54 Mbit/s	2-54 Mbit/s		
Bande	ISM 2,4 GHz	ISM 2,4 GHz	ISM 2,4 GHz	UNII 5GHz	ISM ou UNII		
Couche PHY	FHSS ou DSSS	DSSS-highRate	OFDM	OFDM	DSSS ou OFDM		
QoS	Non	Non	Non	Non	Oui	Non	Non
Handover	Non	Non	Non	Non	Non	Oui	Non
Sécurité	WEP	WEP	WEP	WEP	WEP	WEP	AES

Tableau 4 - Versions 802.11

8. Références

[1] E.Royer, C-K. Toh “A Review of Current Routing Protocols for ad hoc Mobile Wireless Networks" IEEE Personal Communications, Apri11999.

[2] E. Ermel "Techniques et algorithmes de routage dans les réseaux ad hoc "

[3] Josh Broch, David A. Maltz, David B. Johnson, Yih-Chun Hu, Jorjeta Jetcheva " A Performance Comparison of Multi-Hop Wireless ad hoc Network Routing Protocols " In Proceedings of Mobicom '98

[4] Tony Larsson and Nicklas Hedman " Routing protocols in wireless ad-hoc networks - a simulation study" Master's thesis, Lule University of Technology. December 1998.

[5] Navid Nikaein, Shiyi Wu, Christian Bonnet and Houda Labiod "DESIGNING ROUTING PROTOCOL FOR MOBILE AD HOC NETWORKS" Institut Eurecom

[6] IETF MANET WG (Mobile Ad hoc NETwork) www.ietf.ora/html.charters/manet-charter.html

"ad hoc On Demand Distance Vector (AODV) Routing"

[7] IETF MANET WG " Temporally-Ordered Routing Algorithm (TORA) Version 1 Functional Specification "

[8] IETF MANET WG " The Dynamic Source Routing Protocol for Mobile ad hoc Networks "

[9] IETF MANET WG" Optimized Link State Routing Protocol "

[10] D. Johnson, D. Maltz " Dynamic source routing in ad hoc wireless networks, in Mobile Computing " T.Imielinski and H. korth, Eds. Norwell, MA: Kluwer, 1996.

[11] C. Perkins " Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers "

[12] Charles E. Perkins and Elizabeth M. Royer " Ad hoc On-Demand Distance Vector Routing " *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, New Orleans, LA, February 1999, pp. 90-100.

[13] M. Pearlman, Z. Hass. “Determining the optimal configuration for the zone routing protocol”, IEEE selected area in communication, August, 1999.

[14] Shree Murthy, J. J. Garcia-Luna-Aceves " WRP - An Effecient Routing Protocol for Wireless Networks"

[15] C. Chaudet, “Qualité de service et réseaux ad-hoc – un état de l’art”, Rapport de recherche INRIA, RR-4325, Novembre 2001.

[16] S. Chen and K. Nahrstedt, "Distributed quality-of-service routing in ad hoc networks", *IEEE Journal on Selected Areas in Communications, special issue on Wireless ad hoc Networks,* 17(8) :1488{1505, *august* 1999.

[17] R.L. Chunhung and L. Jain-Shing, "QoS routing in ad hoc wireless networks", *IEEE Journal on Selected Areas in Communications, August* 1999.

[18] C.Zhu, M.Scott Corson, "QoS Routing for Mobile ad hoc Networks", Technical Research Report.

[19] K.Wu, I.Harms, "QoS Support in Mobile ad hoc Networks".

[20] D.Chalmers, M.Sloman, "A Survey of Quality of Service in Mobile Computing Environments", IEEE Communications Surveys, Second Quarter 1999.

[21] T.-W. Chen, J.T. Tsai, M. Gerla, "QoS routing performance in a multi-hop, multimedia, wireless networks" , in Proc. IEEE ICUP97 part2, 1997, pp.557-451.

[22] Hannan Xiao Winston "A Flexible Quality of Service Model for Mobile Ad-Hoc Networks"

[23] Samarth H. Shah "Predictive Location-Based QoS Routing in Mobile ad hoc Networks"

[24] Satyabrata Chakrabarti, and Amitabh Mishra. "QoS Issues in ad hoc Wireless Networks" *IEEE Communication Magazine*, February 22001

[25] Chunhung Richard Lin and Chung-Ching Liu "An On-Demand QoS Routing Protocol for Mobile ad hoc Networks"

[26] Amit Kumar Saha "Quality of Service in Dynamic Source Routing (DSR) "

[27] Roy Leung, Jilei Liu, Edmond Poon, Ah-Lot Charles Chan, Baochun Li "MP-DSR: A QoS-aware Multi-path Dynamic Source Routing Protocol for Wireless Ad-Hoc Networks" - D*epartment of Electrical and Computer Engineering University of Toronto*

[28] S. Sheng, “Routing Support for Providing gauranteed end-to-end Quality of Service”,PH.D Thesis, University of IL at Urbana Champaign, http://cairo.cs.uiuc.edu/papers/SCthesis.ps, 1999.

[29] Raghupathy Sivakumar , Prasun Sinha, Vaduvur Bharghavan *" CEDAR: Core Extraction Distributed Ad hoc Routing* , IEEE Journal on Selected Areas in Communication, Special Issue on Ad hoc Networks, Vol 17, No. 8, 1999.

[30] I. Aad and C. Castelluccia, “Differentiation mechanisms for IEEE 802.11”. In to appear in IEEE Infocom 2001, april 2001.

[31] A. Veres, Campbell, A. T, Barry, M and L-H. Sun, "Supporting Service Differentiation in Wireless Packet using Distributed Control", *IEEE Journal of Selected Areas in Communications (JSAC)*, Special Issue on Mobility and Resource Manegement in Next-Generation Wireless Systems, Vol. 19, No. 10, pp. 2094-2104, October 2001

[32] Elizabeth M. Royer, Sung-Ju Lee and Charles E. Perkins. "The Effects of MAC Protocols on Ad hoc Network Communications." *Proceedings of the IEEE Wireless Communications and Networking Conference*, Chicago, IL, September 2000.

[33] Samir R. Das, Charles E. Perkins, Elizabeth M. Royer and Mahesh K. Marina. "Performance Comparison of Two On-demand Routing Protocols for Ad hoc Networks." *IEEE Personal Communications Magazine* special issue on Ad hoc Networking, February 2001, p. 16-28.

[34] Piyush Gupta and P. R. Kumar *" A system and traffic dependent adaptive routing algorithm for ad hoc networks "* , Proceedings of the 36th IEEE Conference on Decision and Control, pp. 2375--2380, San Diego, Dec. 1997.

[35] Cansever, D.H.; Michelson, A.M.; Levesque, A.H. “*Quality of service support in mobile ad-hoc IP networks”*. MILCOM 1999. IEEE

[36] M Kazantzidis “End-to-end versus Explicit Feedback Measurement in 802.11 Networks” - TECHNICAL REPORT # 010034 UCLA Computer Science WAM Lab

[37] M Kazantzidis, M Gerla, "Permissible Throughput Network Feedback for Adaptive Multimedia in AODV MANETs" ICC 2001.

[38] Elizabeth M. Royer and Charles E. Perkins. "An Implementation Study of the AODV Routing Protocol." *Proceedings of the IEEE Wireless Communications and Networking Conference*, Chicago, IL, September 2000.

[39] Gahng-Seop Ahn, Andrew T. Campbell, Andras Veres and Li-Hsiang Sun, "SWAN: Service Differentiation in Stateless Wireless Ad Hoc Networks", Proc. IEEE INFOCOM'2002, New York, New York, June 2002.

[40] Lee, S.B., Gahng-Seop, A., Zhang, X., and A.T. Campbell, "INSIGNIA: An IP-Based Quality of Service Framework for Mobile Ad Hoc Networks",*Journal of Parallel and Distributed Computing (Academic Press)*,Special issue on Wireless and Mobile Computing and Communications, Vol. 60 No. 4pg. 374-406, April 2000.