QOLSR : Routage avec QoS dans OLSR

A. Munaretto[1,2], H. Badis[2,3], K. Al Agha[2,3] et G. Pujolle[1]

[1] *Laboratoire LIP6,Université de Paris VI, France*
[2] *Laboratoire LRI, Université de Paris XI, France*
[3] *INRIA Rocquencourt, France*

Cet article présente QOLSR, un protocole de routage avec QoS pour les réseaux ad hoc. L'algorithme proposé est intégré au protocole OLSR (Optimized Link State Routing) et définit une métrique plus adaptée que le nombre de sauts. Nos simulations montrent que QOLSR fournit des meilleures performances que le protocole OLSR classique.

Keywords: MANET, Routage avec QoS, OLSR, métrique délai

1 Introduction

La qualité de service (QoS) dans l'Internet a été largement étudiée ces dernières années. Dans [XX99], les auteurs en présentent un excellent état de l'art. Cependant, ces travaux appliqués aux réseaux ad hoc conduisent à des performances limitées à cause des contraintes de bande passante et de la topologie dynamique des réseaux MANETs (Mobile Ad Hoc Networks). Pour supporter la QoS, les caractéristiques du lien (le délai, la bande passante, le coût, le taux de perte, le taux d'erreur dans le réseau, etc.) doivent être pris en compte. Toutefois, l'obtention et la gestion de ces caractéristiques constituent un défi car un lien sans fil peut subir des changements provoqués par les limitations des ressources et par la mobilité des nœuds [SC01]. Par conséquent, les protocoles de routage pour les réseaux ad hoc doivent être adaptatifs aux changements fréquents de topologie provoqués par la mobilité des nœuds. Ces changements rendent l'information sur l'état du réseau obsolète. Plusieurs protocoles ont été proposés pour résoudre ce problème [ER99]. Le nombre de sauts est souvent la seule métrique adoptée par les protocoles de routage proposés dans le groupe de travail MANET de l'IETF. Dans un réseau ad hoc, la communication physique se passe par le biais d'une interface radio. Dans ce cas, le nombre de sauts n'est pas nécessairement la métrique la plus adaptée pour être choisie par une décision de routage. Un protocole de routage proactif, tenant compte des caractéristiques du réseau, devient plus approprié dans la sélection de la meilleure route parmi les multiples chemins disponibles. La sélection de la route doit considérer les conditions actuelles des liens. La fonction élémentaire du routage avec QoS est de sélectionner un chemin qui satisfait les contraintes de qualité préalablement spécifiées. Le routage par l'état de lien (e.g. OLSR) amène à l'information détaillée sur la connectivité et les conditions trouvées dans le réseau. En conséquence, il augmente la probabilité qu'un nœud puisse fournir une route avec les contraintes de qualité désirées.

Ce papier propose QOLSR, un protocole de routage avec QoS pour les réseaux ad hoc. QOLSR inclut un paramètre pour améliorer la qualité de service tout en prenant compte des ressources limitées dans un environnement dynamique. Afin d'analyser les performances du protocole, nous nous sommes basés sur le protocole de routage OLSR [TC03], qui considère comme métrique le nombre de sauts.

Cet article est organisé de la façon suivante. Dans un premier temps, nous allons décrire le protocole de routage OLSR puis présenter QOLSR, une extension d'OLSR pour la QoS. Nous évaluerons ensuite les performances du protocole QOLSR en fonction de celles obtenues dans un OLSR classique. Enfin, la dernière partie présentera les conclusions de nos travaux.

2 OLSR

OLSR est un protocole de routage proactif pour les réseaux ad hoc. Ce protocole propose une optimisation permettant de réduire la taille des messages de contrôle et de minimiser l'inondation du trafic de contrôle. Le protocole hérite de la stabilité de l'algorithme d'état de lien et il a l'avantage d'avoir des routes immédiatement disponibles dès que nécessaire grâce à sa nature proactive. OLSR est une adaptation du protocole d'état de lien classique aux les réseaux ad hoc.

OLSR est un protocole proactif qui permet l'échange régulier d'informations sur la topologie entre les nœuds du réseau. Le concept principal utilisé dans le protocole est le relais multipoint (MPR). Les MPRs sont des nœuds sélectionnés qui acheminent les messages diffusés pendant le processus d'inondation. L'idée du MPR est de minimiser l'inondation du trafic de contrôle dans un réseau en réduisant les retransmissions dupliquées dans la même région. Chaque nœud dans le réseau sélectionne un ensemble de nœuds de son voisinage auxquels ses messages seront transmis. Un nœud sélectionne ses MPRs parmi ses voisins à un saut avec un lien symétrique. Cet ensemble est choisi de manière à couvrir (par rapport à la portée radio) tous les nœuds qui sont à deux sauts.

Les nœuds sélectionnés comme MPRs annoncent régulièrement leur condition de MPR dans les messages de contrôle envoyés à son voisinage. De cette façon, un nœud annonce au réseau qu'il est capable d'atteindre les nœuds qui l'ont élu comme MPR. Dans le calcul de la route, les MPRs sont utilisés pour la mise en place des routes d'un nœud donné jusqu'à toutes les destinations du réseau. Ainsi, en sélectionnant la route par l'intermédiaire des MPRs, on évite les problèmes liés à la transmission de paquets sur des liens unidirectionnels. Chaque nœud maintient l'information sur ses voisins qui ont été sélectionnés comme MPR. Un nœud obtient cette information par les messages de contrôle reçus périodiquement de ses voisins. Dans notre proposition, nous utilisons ce même mécanisme de découverte de voisinage pour l'établissement des routes avec QoS.

3 QOLSR

Pour établir une session, un protocole de routage avec QoS doit trouver une route avec les contraintes de qualité spécifiées. Une exigence difficile à répondre serait de trouver, par exemple, la bande passante maximale disponible d'une route (problème NP-complet [ZW96]). Dans [CZ02], les auteurs proposent des garanties de QoS dans le cadre des protocoles de routage réactifs, à l'occurrence AODV [CP03]. Néanmoins, cette proposition ne considère pas le cas où multiples routes QoS sont configurées simultanément.

Pour offrir de la QoS, des extensions seront ajoutées aux messages OLSR pendant la découverte du voisinage. Les messages de contrôle d'OLSR servent à apporter la qualité de service désirée. Chaque nœud doit identifier les interfaces des ses voisins avec lesquelles il a un lien direct et symétrique. Les variations de la propagation radio peuvent rendrent les liens unidirectionnels. Par conséquent, tous les liens doivent être vérifiés dans les deux directions pour devenir valides. Pour accomplir la détection du voisinage, chaque nœud diffuse les messages Hellos qui contiennent des informations sur les interfaces des voisins et l'état de leurs liens. Les messages Hellos sont diffusés à tous les voisins à un saut et sont émis à chaque interface du nœud.

OLSR offre des extensions permettant d'améliorer la fonctionnalité du protocole originel. Ces extensions sont introduites de manière à rester compatible avec des versions antérieures. Des extensions possibles sont le mode d'opération en veille, le routage multicast, et la qualité de service.

La QoS est implémentée dans le protocole OLSR classique en ajoutant des paramètres de qualité de service dans la table de routage. Pour implémenter le routage avec QoS, au moins un paramètre doit être considéré : le délai maximal. Chaque nœud maintient une table de routage pour router les données destinées aux autres nœuds dans le réseau. Le calcul de la table de routage est basé sur les informations contenues dans la table des voisins découverts et dans la table de la topologie du réseau. Ainsi, si l'une de ces tables est modifiée alors la table de routage doit être recalculée afin de faire la mise à jour de l'information de la route disponible pour chaque destination du réseau. Fondamentalement, il existe un nombre de routes possibles entre deux nœuds qui communiquent et chaque chemin doit avoir une capacité disponible différente. Cette capacité peut ou non satisfaire la QoS désirée. Même si le chemin sélectionné entre la source et la destination, répond aux besoins de l'utilisateur pendant le temps d'une session, la capacité et les erreurs observées au long du chemin sont susceptibles de varier dans le temps à cause de la dynamicité du réseau. Une vérification régulière est nécessaire étant donné que les conditions du lien peuvent varier dans le temps [HX00]. On peut donc inclure la fonctionnalité des services adaptatifs, comme par exemple le re-routage après la mobilité des nœuds ou après une dégradation d'une route.

3.1 Métrique pour un environnement ad hoc

Pour améliorer la qualité de service d'un réseau ad hoc, il s'avère pertinent d'inclure des paramètres tels que le délai, la bande passante, le coût du lien, la perte de paquets et le taux d'erreurs. Cependant, [EC98] nous montre que le calcul optimal d'une route prenant en considération deux métriques ou plus est NP-complet. QOLSR inclut le paramètre de délai pour chaque entrée associée à une route dans la table de routage. L'idée est d'ajouter l'information du délai mesuré entre un nœud et son voisin. Néanmoins, mesurer le délai d'un aller simple est normalement une tâche difficile dû au temps de synchronisation nécessaire. Dans notre approche, nous avons supposé un réseau synchronisé. La synchronisation dans les réseaux Ad Hoc présente un axe de recherche très vaste [KR01].

3.2 La métrique délai

Dans notre proposition, le délai sera utilisé pour le choix du meilleur chemin. Chaque nœud inclut dans son message de Hello, pendant la découverte du voisinage réalisé par OLSR, le moment de création de ce message. Quand le message de Hello est reçu par le nœud voisin, le délai entre le nœud émetteur et le nœud récepteur est calculé ($\mathcal{D}_{mes}$). Après, on va calculer la moyenne et l'écart type par l'algorithme suivant :

Soit $\mathcal{D}_{moy}$ le délai moyen de transmission d'un message hello. A chaque réception d'un message hello, l'algorithme suivant est exécuté :

1. $\mathcal{D}_{moy} = \frac{\mathcal{D}_{moy}+\mathcal{D}_{mes}}{i}$
2. $s = s + \mathcal{D}_{mes}^2$
3. $\sigma^2 = abs[(s/i) - \mathcal{D}_{moy}^2]$
4. $i{+}{+}$

où σ^2 est la variance et i est le nombre d'échantillons.

L'information sur le voisinage sera stockée dans la table de voisins, de la même façon que dans OLSR standard. Cependant, QOLSR ajoute le délai nécessaire pour qu'un paquet envoyé par le nœud émetteur atteigne le nœud récepteur. Cette procédure sera exécutée pour tous les messages du type Hello sans inclure des nouveaux messages dans le protocole de routage OLSR.

Pour éviter que les oscillations du réseau soient prises en compte, la valeur du délai inclut l'écart type dans l'analyse statistique. Les nœuds MPRs propagent les informations sur le délai moyen et l'écart type dans les messages TCs. Au moment de calculer la table de routage en utilisant un algorithme de plus court chemin, on vas tester l'écart type par rapport à sa moyenne pour exprimer la stabilité des liens. Par exemple, si le délai actuel est de 19 ms et la valeur antérieure était de 1ms, le délai moyen est de 10 ms, mais la variation de ces valeurs doit être considérée. On va prendre les délais moyennes comme des poids au niveau des arcs.

4 Évaluation des performances

Etant donné qu'un lien ne connaît pas l'état de fonctionnement des ses voisins, on introduit le paramètre de délai dans les nœuds routeurs et non pas sur les liens de routage, dans la mesure où une émission sur un lien engendre des interférences sur les liens avoisinants. Le but de cette section est de tester et d'analyser les performances de QOLSR avec OLSR. Pour ce faire, nous introduisons le paramètre de délai dans le modèle de simulation d'OLSR que nous avons proposé dans des travaux précédents [HB02]. Ce modèle peut simuler jusqu'à 100000 nœuds avec des larges durées de simulation. Nous présentons l'implémentation de notre protocole de routage avec QoS ainsi les résultats obtenus par rapport à l'OLSR classique.

4.1 Implémentation de l'algorithme

Afin d'évaluer les performances de notre schéma, l'implémentation de l'algorithme suis la représentation illustrée dans la figure 1.

Les étapes permettant de calculer la métrique délai sont les suivantes :

- Avant que la couche OLSR envoie un message Hello, elle introduit la date de la création de ce message.

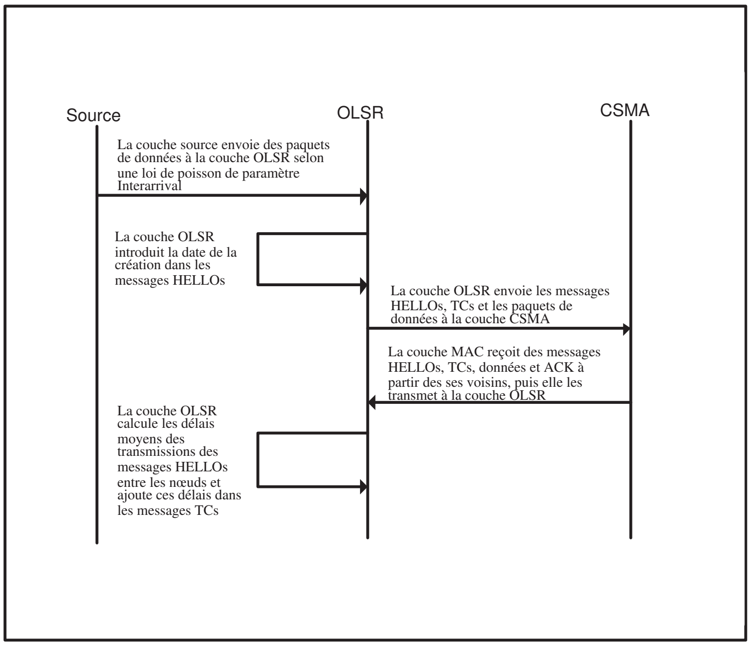

Fig. 1: Le schéma d'implémentation.

- Quand la couche OLSR reçoit un message Hello, elle récupère la date de la création du message envoyé par le nœud émetteur, puis elle calcule la différence par rapport la date actuelle qui représente le délai de transmission d'un aller simple. Une fois le délai calculé, la couche OLSR peut obtenir le délai moyen et l'écart type qui représente la stabilité des liens.

- Pendant le calcul des routes, si le nœud émetteur est un MPR, l'écart type sera examiné. Dans ce cas-là, si l'écart type est inférieur à un certain seuil, afin de maintenir la cohérence des valeurs des délais utilisés, le poids de ce nœud MPR sera son délai moyen. Sinon, son poids sera positionné à 1 seconde qui représente la priorité la plus faible. Ceci permet d'éviter que ce MPR soit sélectionné dans une route.

Dans notre implémentation du protocole de routage OLSR classique, nous avons utilisé l'algorithme de Dijkstra qui est couramment utilisé dans les techniques de découverte de routes d'une source vers une ou plusieurs destinations. Pour avoir des routes optimales en terme de nombre de sauts, il suffit de prendre le même poids (par exemple un poids égal à 1) au niveau des arrêtes du graphe construit à partir de la table des voisins et de la table de topologie.

Cependant, avec QOLSR, le poids est affecté au nœud au lieu de l'arrête. Le poids sera le délai moyen en prenant en considération l'écart type. Avec cette méthode, la métrique de délai et le nombre de sauts sont couplés pour avoir une meilleure route. Nous analysons seulement la valeur de délai des nœuds MPRs pour respecter le protocole OLSR standard qui utilise uniquement les nœuds MPRs pour conduire les paquets.

4.2 Résultats de simulation

Nos simulations sont effectuées par le simulateur OPNET [opn]. Nous évaluons deux scénarios différents. Les valeurs des paramètres utilisés sont : le seuil d'écart type est fixée à 10, la zone de couverture de chaque nœud est de 60 mètres, le seuil de la réception est de 0,00027 watts et le temps d'interarrivé est de 60 s. Ce temps sépare deux générations successives de données pour le même nœud.

Dans le premier scénario, nous analysons leur temps moyen de transmission par rapport au nombre de nœuds dans une région de 10000 m2. Afin d'analyser toutes les configurations possibles de notre protocole, nous avons considéré des nœuds fixes. Nous avons supposé une bande passante de 10kb/s. La figure 2 montre le temps moyen de transmission entre deux nœuds.

Le graphique montre l'amélioration due à l'utilisation de la métrique de délai par rapport au protocole OLSR classique en ce qui concerne le temps moyen de transmission, c-à-d. le temps pour qu'un paquet

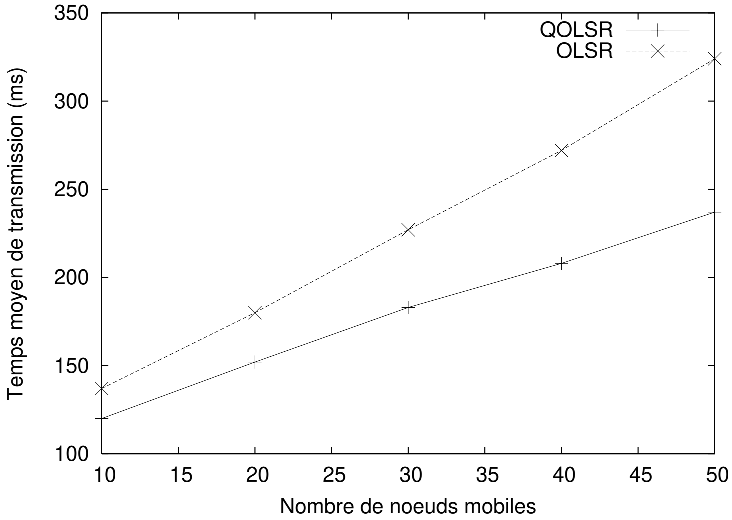

Fig. 2: Le temps moyen de transmission entre deux nœuds.

de données arrive à sa destination finale. Nous constatons que le temps de réception moyen augmente de manière proportionnelle au nombre de nœuds. Ceci est dû aux collisions.

Le prochain scénario évalué présente le comportement de QOLSR proposé dans un réseau avec un taux de collisions élevé. La figure 3 montre que QOLSR fournit de meilleur temps de transmission par rapport à l'OLSR classique. Le temps de transmission augmente en fonction du nombre de nœuds mais avec des gains plus important dans un réseau dense. On constate un gain de 125 ms avec 40 nœuds et de 160 ms avec 50 nœuds.

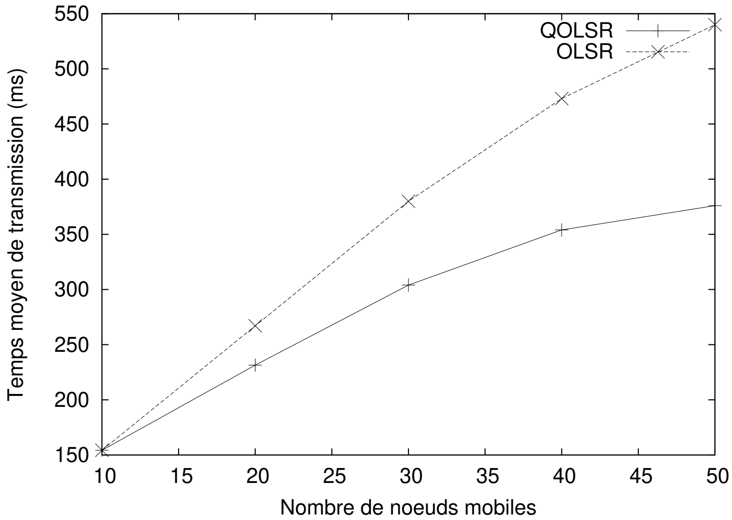

Fig. 3: Le temps moyen de transmission sans mobilité avec un taux de collisions élevé.

Les résultats présentés nous permettent d'observer que, dans la pratique, le nombre de collisions est très lié aux interférences dans le support radio. En plus, le gain obtenu réduit considérablement la période de transmission dans les réseaux sans fil. Le gain obtenu dans le deuxième scénario atteint environ 18%.

5 Conclusion

Dans cet article, nous avons présenté QOLSR, un protocole de routage avec QoS pour les réseaux ad hoc qui utilisent le délai comme métrique. Nous l'avons implémenté en utilisant seulement les messages déjà existants, comme Hello et TC dans le protocole OLSR classique. L'évaluation des performances présentée dans ce papier montre l'amélioration acquise par l'utilisation de notre protocole par rapport à l'OLSR classique. Les gains obtenus sont très conséquents.

Dans une prochaine étape, nous visons à tester et à analyser les performances de QOLSR en considérant la mobilité des nœuds, et puis introduire la métrique de la bande passante.

References

[CP03] S. Das C. Perkins, E. Belding-Royer. Ad hoc on-demand distance vector (aodv) routing. *MANET Working Group, IETF*, 2003. draft-ietf-manet-adov-12.txt.

[CZ02] S. Corson C. Zhu. Qos routing for mobile ad hoc networks. *IEEE Infocom*, June 2002.

[EC98] B. Jajagopalan H. Sandick E. Crawley, R. Nair. A framework for qos-based routing in internet. *IETF*, August 1998. RFC 2386.

[ER99] C-K. Toh E. Royer. A review of current routing protocols for ad-hoc mobile wireless networks. *IEEE Personal Communications Magazine*, pages 46–55, April 1999.

[HB02] K. Al Agha H. Badis. Un modele generique pour la simulation du protocole de routage olsr. *DNAC*, 2002.

[HX00] A. Lo K. C. Chua H. Xiao, W. K. G. Seah. A flexible quality of service model for mobile ad-hoc networks. *IEEE Vehicular Technology Conference*, May 2000.

[KR01] E. Zurich K. Romer. Time synchronization in ad hoc networks. *ACM Mobicom*, 2001.

[opn] Opnet simulator. http://www.opnet.com.

[SC01] A. Mishra S. Chakrabarti. Qos issues in ad hoc wireless networks. *IEEE Communications Magazine*, February 2001.

[TC03] A. Laouiti P. Minet P. Muhlethaler A. Qayyum L. Viennot T. Clausen, P. Jacquet. Optimized link state routing protocol. *MANET Working Group, IETF*, March 2003. draft-ietf-manet-olsr-08.txt.

[XX99] L. M. Ni X. Xiao. Internet qos: A big picture. *IEEE Network*, 13(2), March/April 1999.

[ZW96] J. Crowcroft Z. Wang. Quality of service routing for supporting multimedia application. *IEEE JSAC*, September 1996.