

Impact de la Charge de Contrôle de Routage Sur la QoS dans un Réseau Ad hoc.

Chouaib BOULKAMH, Mohamed Nadjib OUNES, Khaled LAGGOUN, Maamar SEDRATI, Azeddine BILAMI
Département d'informatique. Université de Batna
cboukamd@yahoo.com, msedrati@gmail.com, abilami@yahoo.fr

Résumé- Le but de ce travail est d'étudier et d'évaluer l'influence du trafic de contrôle, généré par le protocole AODV, pour le routage des paquets dans les réseaux mobiles ad hoc. On vise à améliorer la QoS (Qualité de Service) en minimisant le nombre de paquets de contrôle générés par le protocole AODV, pour la recherche, l'établissement et le maintien des routes. Des simulations sous Network Simulator2 (ns2), ont permis de mesurer et de comparer l'influence de ce trafic sur la QoS, dans la version originale du protocole et dans une version modifiée que nous présentons dans cet article.

I. INTRODUCTION

Le protocole AODV (Ad hoc On-Demand Distance Vector Routing Protocol), est un protocole de routage réactif conçu par Charles E. Perkins et Elizabeth M. Royer [1]. Il peut être considéré comme une combinaison des deux protocoles DSDV et DSR, car il détient de DSR ses mécanismes de découverte et de maintenance de routes "Route Discovery" et "Route Maintenance" ; et de DSDV, son routage par sauts "hop by hop", ses numéros de séquences ainsi que la diffusion des mises à jour des tables de routage. Il est basé sur le principe de routage à vecteur de distance.

Au vu de ses caractéristiques, ce protocole est devenu très connu et a fait l'objet d'un grand nombre de recherches. Il est tout à fait adapté aux réseaux mobiles ad hoc de part sa prise en charge de la mobilité des nœuds dans le réseau. Ce protocole permet à des nœuds mobiles d'obtenir des routes rapidement pour les nouvelles destinations, sans maintenir des routes pour lesquelles il n'existe pas de communication active. Ainsi, l'établissement d'une route se fait uniquement en cas de besoin. Si un nœud source veut envoyer des paquets de données vers un nœud destination, il doit établir et maintenir une route vers ce nœud destination durant le temps qu'il en fait usage.

Compte tenu de la nature des MANETs (Mobile Ad hoc Network), il est difficile de supporter des applications en temps réel avec une qualité de service appropriée. Cependant, la prise en charge de la qualité de service est importante dans les MANETs. Elle permet d'améliorer le rendement du réseau dans des applications exigeant une QoS, en offrant la possibilité à l'information essentielle de circuler malgré des conditions difficiles [2].

On se propose d'étudier, d'évaluer et d'optimiser certains paramètres du protocole AODV, très utilisé pour le routage dans les réseaux ad hoc, en considérant le débit des paquets reçus, le débit des paquets perdus, et la quantité de paquets

générés par le protocole AODV pour la recherche, l'établissement et le maintien des routes. Le protocole idéal est celui qui permet de minimiser le nombre de paquets de contrôle et envoyer un maximum de paquets de données utiles tout en réduisant le taux des paquets perdus. Le trafic de contrôle est discuté dans la section deux.

La section trois est consacrée à la description de l'environnement de simulation.

Nous faisons une comparaison entre les deux versions du protocole: AODV et AODV modifié que nous appelons AODV-MC (AODV with Minimum Control), que nous présentons dans la section quatre. Les résultats de la simulation sous NS2 sont présentés et commentés dans la section cinq. Enfin, dans la section six, en conclusion, nous citerons quelques perspectives définissant les travaux futurs que nous comptons mener et qui vont dans le sens de l'amélioration de la QoS dans les réseaux ad hoc.

II. TRAFIC DE CONTROLE

Pour l'établissement et le maintien d'une route le protocole AODV utilise trois types de message : les paquets de demande de route RREQ (Route REQuest Message), les paquets de réponse de route RREP (Route REPLY Message), les paquets d'erreur de route RERR (Route ERRor Message). Ces paquets constituent le trafic de contrôle de routage. Quand un nœud source et un nœud destination n'ont pas une route entre eux, le nœud source initie le RREQ pour découvrir la route, le nœud intermédiaire ou le nœud destination utilise le RREP pour répondre à la requête de découverte de route, le RERR est utilisé dans le cas où il y a une route rompue [3].

A. La stratégie de routage

Le nœud source utilise le paramètre TTL (Time To Live) comme nombre de sauts du RREQ dans le réseau, quand le RREQ est diffusé, le nœud source affecte au TTL la valeur initiale TTL_START et attendra une période RREP_WAIT_TIMEOUT avant d'entreprendre une action. Si une réponse est reçue avant l'expiration de la période d'attente, alors le processus de recherche de route est terminé avec succès. Par contre, si aucune réponse n'est reçue (paquet RREP) durant cette période, le nœud source rediffuse le RREQ et incrémente le TTL d'un TTL_INCREMENT. Il attend alors une période plus longue que la précédente. Avec le TTL augmenté, le RREQ atteint un plus grand nombre de nœuds sur le réseau et a plus de chances de trouver une route (soit il atteint le nœud destination, soit un nœud intermédiaire

ayant dans sa table de routage une route valide vers le nœud destination) [3]. Tant qu'une réponse ne revient pas au nœud source, il continue l'augmentation du TTL et la rediffusion du RREQ jusqu'à un nombre maximum de tentatives RREQ_RETRIES.

Si après RREQ_RETRIES tentatives d'établissement de route, il n'y a aucune réponse alors le processus est abandonné (échec).

Par contre, si une route est établie, alors la valeur TTL aboutissant à cette route sera enregistrée. Elle servira pour des éventuelles recherches futures de routes vers le même nœud destination, ou la valeur TTL_START sera initialisée à la valeur du TTL déjà enregistrée plus un TTL_INCREMENT [3], [4].

III. DESCRIPTION DU SCÉNARIO SIMULÉ

Afin d'étudier et d'analyser l'influence du trafic de contrôle généré par le protocole AODV dans un réseau ad hoc, nous avons utilisé le simulateur Network Simulator2 (ns2) version 2.31 installée sur Debian GNU/ Linux.

A. Modèle de simulation

La simulation est faite sur un réseau ad hoc de six nœuds mobiles en utilisant le protocole AODV pour le routage. Le contexte de notre simulation est illustré dans le tableau ci-dessous.

TABLE I
PARAMETRES DE SIMULATION

Paramètre	Valeur
Temps de simulation	150s
Protocole	AODV
Taille du paquet de données	512
Topographie de simulation	500x500
Taille du buffer (file) des nœuds	50
Vitesse d'émission	5 paquets/s

Un exemple de disposition, état du scénario est visualisé par le NAM (outil d'animation utilisé dans NS) et donné dans la figure 1.

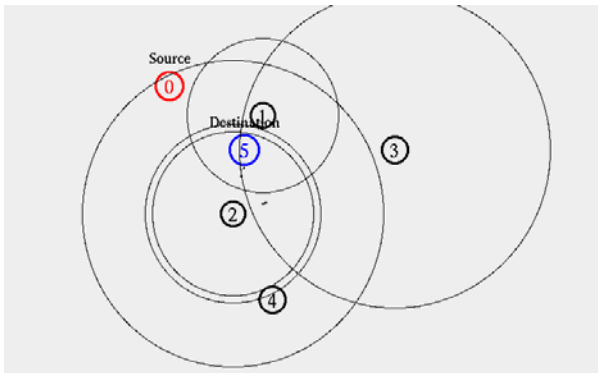


Figure 1 : Exemple de scénario (disposition des nœuds)

Les paramètres standard pour le médium et la propagation radio sont utilisés. Le protocole IEEE 802.11 est utilisé comme protocole d'accès au médium. Le type de gestion de la file ou queue en chaque nœud est 'Drop Tail'. Dans ce type de file, les paquets venant de différents flots sont traités comme de même ordre de priorité: c'est une gestion selon le principe FIFO, c'est-à-dire, une fois que le file se vide, le routeur peut accepter de nouveau des paquets, si le tampon est plein le dernier paquet qui arrive est supprimé. Le nombre maximum des paquets dans le tampon d'émission en chaque routeur est de 50 paquets.

La table II, illustre les paramètres internes du protocole AODV :

TABLE II
PARAMETRES INTERNES AU PROTOCOLE AODV

Paramètres d'AODV	Valeur
Durée de vie des routes dans la table de routage	10
Intervalle des messages HELLO (HELLO_INTERVAL)	1
Nombre de fois que le RREQ est diffusé RREQ_RETRIES	3
TTL_START	5
TTL_INCREMENT	2
Nombre permis de messages HELLO perdus (ALLOWED_HELLO_LOSS)	3

B. Modèle de trafic

Nous avons utilisé des applications qui sont des sources de trafic de type CBR (Constant Bit Rate) et qui émettent des paquets à intervalles réguliers. Ces sources de trafic modélisent la couche application sur des agents de transport UDP. L'utilisation de ce protocole permet d'éviter d'avoir à gérer le contrôle de flux qui amènerait à une analyse plus complexe des résultats. La source (nœud 0 dans notre cas) émet des paquets de taille 512 octets avec un débit de 5 paquets par seconde.

IV. UN AODV MODIFIE : AODV-MC

Comme nous avons déjà dit précédemment le protocole AODV a été étudié intensivement et des propositions de modifications ont fait l'objet de recherches [5], [6], [7]. Dans notre cas, nous nous basons sur l'idée suivante: Si dans la version originale, après RREQ_RETRIES tentatives d'établissement de route, il n'y a aucune réponse alors le processus est abandonné et un message d'erreur est signalé à l'application, nous avons modifié le code source d'AODV de telle sorte de varier le nombre de tentatives en commençant par aucun test ($RREQ_RETRIES = \infty$) sur le nombre de tentatives de diffusion de RREQ. Nous visons par ces tests à définir une valeur adéquate pour chaque scénario, car la non découverte de route peut être aussi due, dans beaucoup de cas, simplement à l'éloignement des nœuds (mobilité). On évitera

ainsi de relancer à chaque fois, l'opération de découverte de route après une certaine période d'attente. Ce qui permet de réduire le trafic de contrôle. Une version modifiée d'AODV, appelée AODV-MC est ainsi définie.

V. COMPARAISON ENTRE AODV ET AODV-MC

Dans cette section nous présentons une comparaison entre L'AODV et L'AODV modifié (AODV-MC), les simulations de ces deux protocoles ont été faites sur le même contexte de simulation, le même modèle de simulation et le même modèle de trafic.

Les figures 2 et 3 représentent respectivement les paquets de contrôle générés dans le cas d'AODV-MC et d'AODV non modifié. Le fait de considérer les courbes dans la période 60 et 100 sec, c'est que durant cette période on a constaté la différence essentielle entre les deux protocoles (dans les autres parties des courbes, il n'existe aucune différence). Dans cette période les nœuds se sont éloignés (plus précisément à partir 80 sec) ce qui signifie que la découverte de route est difficile. Mais il est à noter également que dans cette période les nœuds se déplacent de telle sorte que la route soit par la suite possible à découvrir.

Donc à partir du temps 80 sec, on voit que dans AODV non modifié la courbe décroît, pour augmenter après environ 10 sec. Cela se justifie par le fait que le nœud source (après le test de RREQ_RETRIES tentatives de diffusion de RREQ) attend un certain temps (10 sec) pour initier la procédure de détection de route, tandis que dans la version du protocole AODV-MC qui ne fait pas ce test, on continue l'envoi des paquets de découverte de route.

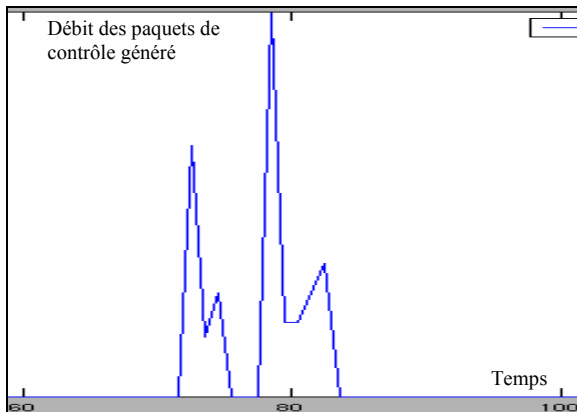


Figure 2 : paquets AODV-MC Générés.

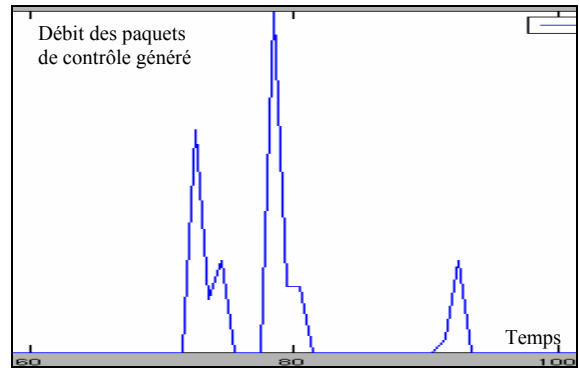


Figure 3 : paquets AODV Générés.

Dans la table III ci-dessous, nous présentons les résultats de la simulation donnant en fonction d'un nombre de paquets générés dans les protocoles AODV et AODV-MC, le nombre de paquets reçus et perdus.

TABLE III
PARAMETRES INTERNES AU PROTOCOLE AODV

PAQUET	AODV			AODV-MC		
	Générés	Reçus	Perdus	Générés	Reçus	Perdus
Total	26144	19555	6589	27974	20942	7032
AODV	228	199	29	230	205	25
Autres	25916	19365	6560	27744	20737	7007

De ce tableau, on constate que : dans le protocole AODV-MC, parmi la totalité des paquets générés, nous avons 0.82% paquets de contrôle tandis que dans le protocole AODV, nous avons 0.87% paquets de contrôle. En outre, le taux des paquets reçus avec succès dans le protocole AODV-MC est 74.85%, et celui des paquets perdus est 25.15%, mais dans le protocole AODV on a 74.79% des paquets reçus et 25.21% des paquets perdus.

Les courbes ci-dessous illustrent le débit des paquets reçus dans les deux protocoles.

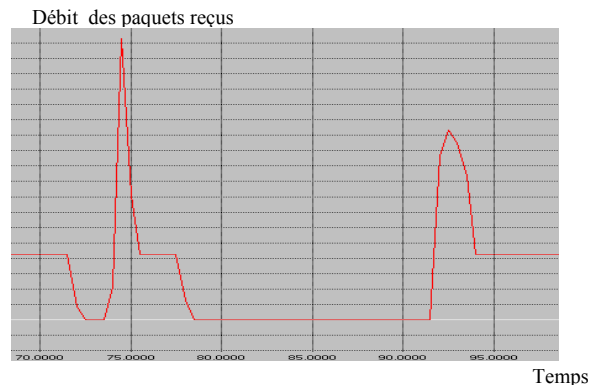


Figure 4: débit des paquets reçus avec AODV.

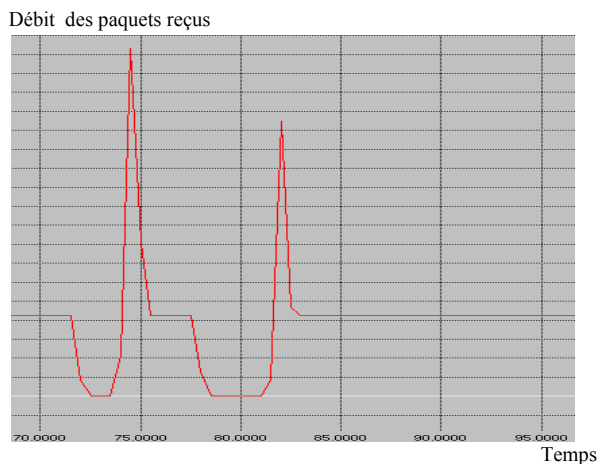


Figure5 : débit des paquets reçus avec AODV-MC

Sur l'intervalle 70 à 80 sec, ces deux courbes sont presque identiques, après cet intervalle le débit commence à diminuer jusqu'à ce qu'il atteigne une valeur minimale, cela est dû à une perturbation du réseau causée par l'éloignement (la mobilité) des nœuds. Dans le protocole AODV-MC, après une courte durée (2sec) le débit commence à augmenter de nouveau, cela signifie qu'une nouvelle route est établie, ce qui n'est pas le cas dans le protocole AODV où le débit n'augmente qu'après une durée relativement longue (>10sec). Ce résultat est justifié par le fait que dans le protocole AODV et après qu'une route soit brisée, le nœud source tente de chercher une nouvelle route, si le nœud échoue après RREQ_RETRIES tentatives (ici 3), il attend une durée avant que le processus de découverte de route soit initié. Par contre dans le protocole AODV-MC, cette limitation n'existe pas, donc le nœud continue le processus de recherche jusqu'à ce qu'une route valide soit trouvée.

VI. CONCLUSION ET PERSPECTIVES

Dans ce travail, nous avons étudié l'effet de la variation de certains paramètres, dans le protocole AODV, sur la qualité de service dans les réseaux ad hoc (essentiellement la perte et le débit utile). On s'est intéressé à la charge du trafic circulant dans ce réseau en évaluant l'impact du trafic de contrôle, le taux des paquets reçus et perdus.

Ce travail toujours en cours, constitue une base pour nos futurs travaux de recherche. Voici les extensions qui nous semblent importantes :

- Définir une valeur optimale pour le paramètre de temps de vie d'une route, ART (Active Route Timeout), en vue d'améliorer le taux de livraison des paquets, PDR (Packet Delivery Ratio). En fait, un ART court, peut causer un nouveau processus de découverte d'une route pour un chemin qui est encore valide, en revanche, le choix d'un ART long, peut conduire à une transmission sur une route invalide et par conséquent, des paquets sont perdus. Donc, il faut trouver un

meilleur compromis entre la valeur de ART et la mobilité des nœuds du réseau ad hoc.

- Elargir l'intervalle de transmission des paquets HELLO, mais sans diminuer le PDR.

- Trouver un mécanisme pour guider la diffusion des messages RREQ, afin d'éviter l'envoi des paquets doublants, ce qui peut optimiser le temps de découverte d'une route.

REFERENCES

- [1] C. Perkins, E. Belding-Royer: "Ad hoc On Demand Distance Vector Routing", the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'99), New Orleans, Louisiana, USA, February 25-26, 1999.
- [2] C. Perkins, E. Belding-Royer, "Quality of Service for Ad hoc On-Demand Distance Vector Routing (work in progress)", Oct 2003, draft-perkins-manet-aodvqos-02.txt. <http://www.reseautoyen.be/wiki/index.php/AODV#Introduction>
- [3] N. Tebbane, S. Tebbane, A. Mehaoua, "Simulation et Mesure des performances du protocole de routage AODV", JTEA'2004, Hamamet, Tunisia 2004
- [4] K. Srdjan, M. Dupcinov, S. Murphy, "Performance Evaluation of an IEEE 802.11b Ad-hoc Network", SER.: ELEC. ENERG. vol. 17, April 2004, 53-68
- [5] S. Krco, M. Dupcinov, "improved neighbor detection algorithm for aodv routing protocol", IEEE Communication Letters, Dec. 2003.
- [6] D. Budke, K. Farkas, O. Wellnitz, B. Plattner, L. Wolf. "Real-Time Multiplayer Game Support Using QoS Mechanisms in Mobile Ad Hoc Networks", In Proceedings of The Third Annual Conference on Wireless On demand Network Systems and Services (WONS 2006), France, January 2006.
- [7] Erik Nordstrom. AODV-UU implementation <http://core.it.uu.se/AdHoc/AodvUUImpl>.