JUMBO : protocole de routage unicast dans les réseaux ad-hoc

Johanne Cohen, Eric Fleury, Jens Gustedt

JUMBO : protocole de routage unicast dans les réseaux ad-hoc.

Johanne Cohen and Eric Fleury and Jens Gustedt [†]

LORIA, Université Henry Poincaré & INRIA, campus scientifique, BP 239, F-54506 Vandœuvre lès Nancy, France

Un réseau ad-hoc est une collection d'entités mobiles, interconnectées par une technologie sans fil formant un réseau temporaire sans l'aide de toute administration ou de tout support fixe. Du fait que le rayon de propagation des transmissions des entités de chacune soit limité, il se peut qu'un sommet se trouve dans l'obligation de demander de l'aide à un autre sommet pour pouvoir communiquer avec son correspondant. Nous présentons un protocole pour réaliser des communications dans les réseaux ad-hoc sans fils de type HIPERLAN. Après une description des contraintes propres à ces réseaux, nous décrivons une fonction de routage pour ces réseaux construite à partir d'une décomposition en sous graphes complets du réseau. Enfin, nous présentons succinctement un algorithme distribué permettant de construire une telle décomposition du graphe.

Keywords: Réseaux ad-hoc, fonction de routage, protocole de communications, algorithme distribué.

1 Modélisation d'un réseau sans fils

L'architecture d'un r´eseau sans fil ad-hoc peut se caract´eriser par une absence d'infrastructure fixe pr´eexistante, `a l'inverse des r´eseaux filaires ou des r´eseaux sans fil tels que nous les connaissons dans les t´el´ecommunications classiques (GSM par exemple). Les nœuds d'un r´eseau ad-hoc sont des hôtes mobiles ayant des capacit´es et une puissance de transmission identique, et poss´edant une certaine puissance de calcul. Un tel r´eseau doit s'organiser automatiquement de façon `a être d´eployable rapidement et pouvoir s'adapter aux conditions de propagation, aux trafics et aux diff´erents mouvements pouvant intervenir au sein des nœuds mobiles.

Du fait de l'absence d'une architecture fixe et de la mobilit´e des nœuds, la topologie d'un r´eseau ad-hoc peut ´evoluer `a tout instant. Afin que le r´eseau reste connecter au sens classique du terme (tout sommet peut atteindre tout autre), chaque sommet est susceptible d'être mis `a contribution pour participer au routage et pour retransmettre les paquets d'un nœud qui n'est pas en mesure d'atteindre directement sa destination. Tout nœud joue ainsi le rôle d'hôte et de routeur. Le groupe de travail de l'IETF [‡] ´etudie ce genre de probl`eme.

Notre ´etude a ´et´e r´ealis´ee en prenant comme type de r´eseau ad-hoc la norme d'HIPERLAN [§], mais les principes des protocoles de routage que nous pr´esentons s'applique aussi `a d'autres r´eseaux ad-hoc.

Une des caract´eristiques primordiale des r´eseaux sans fils est l'utilisation d'un m´edium de communication partag´e qui est le canal de communication hertzien dans notre cas. Une transmission s´elective se r´ev`ele donc impossible : lorsqu'un nœud ´emet, tous ces voisins (*i.e.*, tous les nœuds se trouvant dans la zone de couverture du sommet ´emetteur) vont recevoir le message et une collision se produit si tout autre sommet voisin tente de communiquer en même temps. Pour plus de d´etail sur les r´esolutions de collision et plus g´en´eralement sur HIPERLAN on se r´ef´erera `a [1, 5].

[†] Ce travail a ´et´e en partie financ´e par l'action coop´erative COMPAS de l'INRIA http://menetou.inria.fr/

[‡] http://www.ietf.org

[§] http://www.etsi.org

La topologie d'un r´eseaux ad-hoc ´evoluant `a chaque instant, nous allons la mod´eliser par un graphe $G_t = (V_t, E_t)$ o`u V_t repr´esentent l'ensemble des nœuds et E_t mod´elise l'ensemble des connections existantes entre les nœuds du r´eseau. Il existe une arête $e = (u,v) \in E_t$ si les nœuds u et v sont en mesure de communiquer directement `a l'instant t.

Étant donn´e que tous les sommets utilisent une même fr´equence pour communiquer, lorsqu'un nœud ´emet, il empêche ces voisins d'´emettre au même moment. Cette caract´eristique fondamentale se traduit en la propri´et´e de base suivante valable `a chaque instant t au sein de chaque sous graphe complet (clique) du graphe G_t :

Propriété 1 *Pour toute clique C de G_t :*

- *Au plus un nœud émet un message à l'instant t ;*
- *Lorsqu'un des nœuds de la clique émet un message, tous les autres nœuds de la clique peuvent entendre ce même message.*

Pour g´erer les ´evolutions de la topologie, les nœuds d'un r´eseau HIPERLAN effectuent r´eguli`erement des communications pour signaler leur pr´esence `a leurs voisins. De cette façon, les nœuds ont `a tout moment connaissance des nœuds avec lesquels ils peuvent directement communiquer : la r´eception d'un message « HELLO » signale la pr´esence d'un nœud avec lequel il est possible de communiquer directement ; la non r´eception depuis un certain temps de messages de la part d'un nœud signale la disparition de ce nœud du voisinage, soit parce que le nœud est sortie de la zone de couverture du fait de sa mobilit´e, soit que le lien de communication soit devenu trop faible (trop de bruit, changement des conditions de propagation...). Chaque nœud g`ere localement ces ´ev`enements pour maintenir sa connaissance locale du r´eseau.

2 Fonction de routage

Les hypoth`eses de travail que nous avons faite rejoignent celles ´enonc´ees dans [3]. Le protocole de routage JUMBO est adapt´e aux r´eseaux assez denses dans lequel le trafi c est al´eatoire et sporadique entre plusieurs nœuds du r´eseaux. En effet, JUMBO est un protocole de type « pro-actif » dans le sens o`u il va chercher `a maintenir une vue globale du r´eseau dans chaque nœud afi n que ceux ci puissent router les messages contrairement `a une approche « r´eactive » dans laquelle un nœud va chercher `a d´ecouvrir une route avant de pouvoir communiquer [4].

Nous supposons qu'`a tout instant il existe une **décomposition du réseau en cliques** et que chaque nœud la connaît. Une telle d´ecomposition en cliques est repr´esent´ee par une famille $\mathcal{C} = C_1, \ldots, C_\ell$ de cliques du graphe.

La fonction de routage est alors d´ecrite de la façon suivante :

$$\mathcal{R}(v_{current}, dest) = (C, n_{next}) \tag{1}$$

o`u

- $v_{current}$ est le nœud courant.
- $dest$ est le nœud destination du message.
- C est la clique du graphe qui transmettra l'information `a la prochaine ´etape.
- $n_{next} \in C$ est un nœud de cette clique qui sera choisit de retransmettre l'information.

Cette d´efi nition ne d´ecrit pas un chemin entre deux sommets du graphe compos´e d'arêtes comme il est d'usage dans les r´eseaux fi laires, mais le d´ecrit en terme de clique et de sommet responsable de la retransmission au sein de chacune des cliques. Cette progression d'un message de clique en clique pour aller d'un sommet u `a un sommet v mod´elise le fait que lorsqu'un sommet ´emet, tous les autres sommets de la clique non seulement ne peuvent pas ´emettre mais en plus reçoivent le message. On peut alors appliquer le même genre de raisonnement pour la construction d'un arbre de multicast. Ce qui nous int´eresse ce n'est plus un arbre recouvrant tous les membres du groupe mais bien un arbre recouvrant l'ensemble des cliques o`u se trouve des membres du groupe. Cette notion de d´ecomposition en clique est en fait n´ee de notre volont´e de proposer un protocole de routage multicast et de la constatation que la notion de multi

Algorithme 1: Gestion des ´ev´enements li´es `a la modification de la topologie au niveau du nœud *u*.

```
1  Détecter un événement lié à une modification du réseau ;
   suivant événement faire
2      cas où " v est un nouveau nœud du graphe "
3          Insérer si possible le nœud v dans les cliques dont u est responsable ;
4          Répondre à v en envoyant un message contenant les voisins de u plus les arêtes couvertes par
           insertion éventuelle de v dans les cliques dont u est responsable ;
5      cas où " v est un nouveau voisin de u "
6          Préparer l'émission d'un message qui avertit v de la gestion de cet événement ;
7          si réception du message avertissant u que v gère cette événement avant émission de ce message
           alors
               Répondre à v en renvoyant un message contenant les voisins de u ;
8          sinon
9              Envoi du message ;
10             si réception de la réponse de v alors
11                 insérer v dans les cliques dont u est responsable ;
12                 si besoin alors créer une nouvelle clique réduite à l'arête uv ;
13     cas où " disparition avertie du nœud v "
           Supprimer le nœud v dans la décomposition en cliques ;
14     cas où " manque de HELLO du nœud v "
15         Supprimer le nœud v dans la décomposition en cliques ;
16         Avertir les voisins de u de la suppression temporaire de u dans le réseau ;
17         Déclencher l'événement " insertion d'un nouveau nœud " pour u ;
```

points relais telle qu'elle est définie dans HIPERLAN (les multi points relais d'un sommet u forment un sous ensemble de ses voisins permettant d'atteindre tous les sommets à distance 2 de u) est mal adaptée à la construction d'arbres de multicast.

Nous imposons que la décomposition en cliques du réseau respecte certaines contraintes :

Propriété 2 *L'union des cliques de la décomposition est égale au graphe G_t.*

C'est-à-dire, toute arête est couverte par au moins un élément de la décomposition, et inversement, un élément de la décomposition ne doit pas couvrir des arêtes n'étant pas dans G

Propriété 3 *La décomposition $\mathcal{C} = C_1, \ldots, C_\ell$ du graphe G_t doit vérifier $\ell \leq m$ où $m = |E_t|$ correspond aux nombres d'arêtes de G_t.*

Cette dernière propriété permet d'éviter une gestion trop coûteuse du routage, en raison d'un nombre trop élevé de cliques. Observons que la solution qui consisterait à construire une décomposition du graphe minimale en nombre d'éléments semble elle aussi très coûteuse (le problème MINIMUM CLIQUE COVER est NP-complet [2]). La propriété 3 permet de garantir un bon compromis entre les deux extrêmes.

3 Algorithme de décomposition en cliques

Les nœuds ont une vision locale du réseau : ils connaissent leurs voisins par l'intermédiaire de la réception ou de la non réception de messages HELLO. Pour que l'algorithme de routage s'effectue correctement, il faut toutefois qu'ils puissent en plus connaître à tout moment la décomposition globale du réseau en cliques afin de pouvoir calculer et mettre à jour des routes entre les cliques. Pour résoudre ce problème, nous proposons qu'à chaque clique soit associée un responsable. Ce nœud se charge régulièrement de diffuser les cliques dont il est le responsable à l'ensemble des sommets du réseau (par inondation). Ces diffusions permettent de

garantir que chacun des nœuds du r´eseau connaît `a intervalles de temps r´egulier la d´ecomposition globale du r´eseau en cliques. L'inondation est un moyen simple de diffusion mais il n'est pas `a ´ecarter l'utilisation de techniques plus efficace et moins coûteuse en terme de ressources utilis´ees pouvant se baser sur la d´ecomposition en clique sous-jacente même si cette derni`ere est erron´ee ou incompl`ete.

Nous proposons maintenant un algorithme distribu´e pour construire une d´ecomposition du graphe en sous graphe complets qui v´erifie les propri´et´es 2 et 3.

Continuellement, les nœuds reçoivent des messages de contrôles (incluant les messages `HELLO`). Ces messages permettent de g´erer les ´ev´enements li´es `a la modification de la topologie du graphe (apparition d'un nœud, d´etection d'un nouveau voisin, disparition d'une arête). Les nœuds g`erent ces ´ev´enements dans l'ordre de leurs arriv´ees selon l'algorithme 1 (d´ecrit de façon informelle). Le coût de chaque modification du r´eseau pour r´eactualiser localement la d´ecomposition du graphe vaut en nombres de messages $O(d)$ et en op´erations ´el´ementaires $O(d^2)$ o`u d correspond au nombre maximum de voisins dans le graphe.

Lors d'une insertion d'un nouveau nœud u dans le r´eseau, u d´etermine dans un premier temps son voisinage en ´ecoutant de façon passive ce qui ce passe. Ensuite, il informe tous ces voisins de l'´ev`enement "u est un nouveau nœud du graphe" tout en leur transmettant la connaissance qu'il a de son voisinage. Il attend alors les r´eponses (cf. instruction 2 de l'algorithme) de ses voisins tout en g´erant, si besoin les modifications locales du r´eseau (disparition d'un voisin, apparition d'un nouveau voisin, apparition d'un nouveau nœud du r´eseau dans le voisinage de u). Lors ce qu'il a reçu toutes les r´eponses de ces voisins courants, il cr´ee de nouvelles cliques pour couvrir toutes les arêtes non encore couvertes, et il devient responsable de toutes ces cliques. Si par exemple, pendant cette attente, le nœud u d´etecte une disparition d'un de ces voisins, le nœud r´eactualise les ensembles des nœuds dont il attend une r´eponse.

On d´emontre assez facilement que :

Lemme 1 *La décomposition en cliques du réseau par l'algorithme distribué précédent est une décomposition valide qui vérifie les propriétés 2 et 3.*

4 Conclusion

Dans ce papier, nous avons propos´e un algorithme de routage dans les r´eseaux sans fils ad-hoc de type HIPERLAN bas´e sur une d´ecomposition en cliques du r´eseau. Cette approche se base sur le maintient d'une vue du r´eseaux afin de pouvoir tenir des tables de routage `a jour et ce malgr´e la mobilit´e des hôtes au sein du r´eseau ad-hoc. L'approche de d´ecomposition en clique, permettant de construire des routes est `a l'origine destin´ee `a la construction d'arbres de multicast. Nous allons donc poursuivre dans cette voix mais nous allons aussi tenter de d´eterminer les performances de notre algorithme de routage via des simulations sous `ns` mais via la mise en œuvre d'un prototype au dessus d'IP.

Références

[1] S. K. Barton, editor. *Wireless Personal Communication: Special issue on the High PErformance Radio Local Area Network (HIPERLAN)*, volume 4. Kluwer Academic Publishers, January 1997.

[2] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the theory of NP-Completeness*. a Serie of books in mathematical sciences. Freeman edition, 1979.

[3] P. Jacquet, P. Muhlethaler, and A. Qayyum. Optimized link state routing protocol. Internet Draft draft-ietf-manet-olsr-00.txt, IETF MANET Working Group, November 1998.

[4] D. B. Johnson. Routing in ad hoc networks of mobile hosts. In *IEEE Workshop on Mobile Computing Systems and Applications*, December 1994.

[5] ETSI TC-RES. Radio Equipement and System (res); High PErformance Radio Local Area Network (HIPERLAN), Type 1, Functional specification. Technical Report ETS 300 652, European Telecommunication Standards Institute, December 1995. Draft.