HAL
open science

DRC: Mécanisme de clustering pour la gestion par politiques dans les réseaux ad hoc

Hajer Ferjani, Mouna Ayari

To cite this version:

Hajer Ferjani, Mouna Ayari. DRC: Mécanisme de clustering pour la gestion par politiques dans les réseaux ad hoc. 2006. inria-00113781

HAL Id: inria-00113781

https://inria.hal.science/inria-00113781v1

Submitted on 20 Nov 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

DRC : Mécanisme de Clustering pour la Gestion par Politiques dans les réseaux ad hoc

Hajer Ferjani, Mouna Ayari
Laboratoire CRISTAL ; Ecole Nationale des Sciences de l'Informatique
La Manouba, Tunisie
f.hajer@gmail.com; mouna.ayari@cristal.rnu.tn

RESUME
Les réseaux ad hoc sont caractérisés par une topologie variable et des ressources limitées qui rendent leur gestion particulièrement difficile. Dans ce contexte, la gestion par politiques parait une solution prometteuse pour simplifier les processus de gestion, de configuration et de contrôle de tels réseaux. Cette solution étant basée sur une architecture centralisée, plusieurs travaux ont été menés afin de l'adapter à la nature distribuée des réseaux ad hoc. La technique de regroupement (clustering) a été souvent retenue à cet effet. Dans ce papier, nous proposons un nouvel algorithme de clustering approprié à la gestion par politiques dans les réseaux ad hoc. Notre algorithme permet la formation de clusters de taille et de rayon configurables. La stabilité de ces clusters est améliorée par l'introduction de facteurs de contrôle des réaffiliations. En deuxième étape, nous évaluons par simulations ses performances en fonction de la mobilité des nœuds.

MOTS-CLES
Ad hoc, gestion par politiques, clustering, NS.

1. INTRODUCTION
Les réseaux ad hoc sont des réseaux sans fil spontanés, mobiles et totalement autonomes. En outre, ces réseaux sont caractérisés par une bande passante et une énergie limitées [9]. La gestion de tels réseaux et le déploiement de services avancés tel que le contrôle d'accès ou la qualité de service posent des défis considérables. D'autre part, la gestion des réseaux par politiques [6] est une technique prometteuse qui présente plusieurs bénéfices. Cette solution largement adoptée au niveau des réseaux filaires fixes permet de simplifier et d'automatiser les mécanismes de gestion et de configuration de la qualité de service (QoS), rendant possible la configuration des nœuds en fonction des contraintes du réseau et des besoins des applications sans l'intervention d'un administrateur; ce qui serait particulièrement approprié au contexte des réseaux ad hoc. Dans une telle architecture, la gestion et le contrôle sont centralisés. On dispose d'un serveur de politiques PDP qui gère et administre un ensemble de PEP fixes appartenant à un domaine stable. Une adaptation de cette architecture est donc nécessaire pour étendre son utilisation aux réseaux ad hoc. Plusieurs travaux [14], [16], [2] se sont intéressés à cet aspect. La méthode communément utilisée est la division du réseau en groupes. Dans chaque groupe, un chef est élu pour assumer le rôle de PDP. Les autres membres jouent le rôle de PEP. Nous nous intéressons en particulier à la solution décrite dans [2] qui représente une extension de la gestion par politiques pour le support de la spécification, la négociation, la distribution et l'application des Ad-Hoc-SLA, sorte d'accord entre les nœuds définissant les paramètres et les mécanismes de performance, de contrôle et de configuration liés au service de transport. Nous pensons que les performances du mécanisme de regroupement utilisé pour la sélection des PDP et la formation dynamique des domaines influe considérablement sur la qualité et les performances globales du système de gestion. D'où l'intérêt de la proposition d'un mécanisme de regroupement efficace et adéquat à la gestion par politiques dans les réseaux ad hoc. Dans cet article, nous présentons un nouvel algorithme de clustering DRC (Distributed Reaffiliations Controlled). Il permet de former dynamiquement des clusters à rayon variable. Les PDP ou clusterheads sont choisis parmi les nœuds les plus performants dans le réseau. La stabilité des clusters formés est assurée grâce à l'introduction de contraintes sur les réaffiliations.

Ce papier est organisé comme suit. Dans la section suivante, nous exposons brièvement l'état des lieux des mécanismes de clustering présentés dans la littérature. Dans la troisième section, nous décrivons notre algorithme de clustering DRC, son principe et ses propriétés. La dernière section présente les résultats des simulations menées.

2. ETAT DES LIEUX DES MECANISMES DE CLUSTERING
Le clustering, ou regroupement, a été utilisé pour différents objectifs comme la mise à l'échelle des réseaux ad hoc [13], l'abstraction de la topologie pour le contrôle de l'inondation dans les réseaux [12], la collecte d'informations dans les réseaux de capteurs [11] et le partage de la bande passante [10].

Les premiers algorithmes de clustering Lowest-ID [3], [10] et Mobic [4] ont des mécanismes assez proches. Ils se sont basés sur un critère particulier pour le choix des clusterhead ou chefs de groupe, respectivement les identificateurs des nœuds, le nombre de voisins et le degré de mobilité. Ces algorithmes permettent de former des clusters à un seul saut, où chaque membre est voisin direct de son clusterhead. Ils considèrent une phase de formation des clusters ou « clustering set up ». Pendant cette phase, les nœuds procèdent à la connaissance de leurs voisins et déroulent entre eux l'algorithme de formation des domaines. Toutefois, les nœuds sont supposés fixes au cours de cette étape et une synchronisation entre eux est nécessaire pour le bon déroulement de l'algorithme. De plus, cette phase de formation des clusters est répétée périodiquement suite aux changements de topologie qui peuvent survenir. La ré-exécution périodique de ce processus du clustering dégrade la stabilité des clusters.

Dans [13], les auteurs présentent un mécanisme de clustering qui permet de réduire l'overhead de clustering. Chaque nœud ne diffuse qu'un seul message pendant la phase de formation des domaines. Toutefois, là aussi, l'hypothèse d'absence de mobilité pendant la formation des clusters doit être vérifiée. En outre, le mécanisme de clustering proposé s'affranchit de la notion des clusterhead et ne traite pas le cas où ces derniers quittent le cluster.

L'algorithme "Distributed and Mobility Adaptive Clustering" présenté dans [5] et [15] a introduit la notion de poids générique pour la sélection des clusterhead. C'est un mécanisme de clustering qui permet de réagir aux changements de topologie. L'algorithme ne nécessite aucune synchronisation entre les nœuds. Pour améliorer la stabilité des clusters formés, deux nouveaux facteurs de performances ont été définis. Le premier, K, autorise au maximum K clusterheads à être voisins directs. Le deuxième, H, permet de limiter les réaffiliations entre les clusters. Les nœuds ne se réaffilient à un nouveau clusterhead que lorsque le poids de ce dernier est supérieur d'un certain facteur H au poids de leur clusterhead courant. Toutefois, cette solution ne permet la formation que de clusters à un saut et le facteur de performance H est difficile à spécifier de façon judicieuse.

Dans [8], les auteurs ont présenté une formule multi-critères pour les choix des clusterhead. Elle prend en considération la mobilité, la connectivité et l'énergie disponible. Ce mécanisme de clustering "Weighted Clustering Algorithm" nécessite, toutefois, une synchronisation globale et un échange de voisinage entre tous les nœuds du réseau.

Dans d'autres travaux [7] et [1], les auteurs ont essayé de présenter des algorithmes adéquats à la formation de clusters à k sauts. Toutefois, [1] gère la mobilité par ré-exécution périodique de tout l'algorithme. [7] nécessite d'une part des informations sur le k voisinage et d'autre part que les nœuds vérifient l'hypothèse de non mobilité pendant la phase de clustering.

[16] présente un mécanisme de clustering basé d'une part sur la connaissance préalable de l'aire de déploiement du réseau et sur la capacité de se positionner et d'autre part sur la prédiction des mouvements des nœuds en considérant leur historique.

3. L'ALGORITHME DE CLUSTERING DRC

Dans le contexte de la gestion par politiques, il est nécessaire, pour la fiabilité des services déployés, que tous les nœuds du réseau soient à tout moment attachés à un cluster (ou clusterhead). Ainsi, les algorithmes de clustering gérant la mobilité par ré-exécution périodique du processus de clustering ne sont pas adéquats. En outre, la ré-exécution périodique du processus de clustering aboutit à la destruction des clusters en place et l'élection de nouveaux clusterhead. Or, tout changement d'un cluster à un autre génère un trafic important de synchronisation des politiques entre l'ancien et le nouveau PDP. Par ailleurs, plusieurs des algorithmes de clustering présentés dans la littérature permettent la formation de clusters à un seul saut. Ceci peut être avantageux dans le cas de négociations intra clusters (entre PDP et PEP). Mais, ceci peut augmenter le nombre de clusters générés dans le réseau et par suite le délai des négociations inter clusters (entre PDP).

Dans ce qui suit, nous présentons notre nouvel algorithme de clustering, nommé « Distributed Reaffiliations Controlled Clustering Algorithm ». Il a été conçu pour répondre aux besoins de la gestion par politiques dans les réseaux ad hoc.

Les clusters sont formés suite à l'élection des clusterhead et les réaffiliations des nœuds à ces clusterhead. Nous supposons que chaque nœud dans le réseau ad hoc implémente les fonctionnalités des PDP et PEP et peut ainsi changer de rôle à tout moment de déroulement du processus de clustering.

3.1 Principe

Notre algorithme de clustering est basé sur les principes suivants : la désignation des clusterhead, la formation des clusters et la maintenance des clusters formés.

3.1.1. Désignation des clusterhead

Le mécanisme de désignation des clusterhead permet de sélectionner les nœuds ayant les meilleures performances pour jouer le rôle de PDP. Nous nous sommes inspirés des algorithmes [5] et [8] pour définir une métrique multi-facteurs qui servira pour le calcul des poids. Plus le poids d'un nœud est élevé, plus le nœud est adapté au rôle de clusterhead. Cette métrique prend en considération la mobilité, l'espace de stockage, l'énergie disponible et le degré de connectivité.

3.1.2. Formation des clusters

La formation des clusters se fait par l'élection des clusterhead et par la réaffiliation des nœuds à ces clusterhead. Contrairement à beaucoup d'algorithmes dans la littérature, le mécanisme d'élection des clusterhead n'est pas synchronisé entre tous les nœuds du réseau. Il n'implique pas que tous les nœuds exécutent en même temps la procédure d'élection. Le principe est inspiré de celui présenté dans [5]. La décision d'être clusterhead est effectuée par chaque nœud ne détectant pas dans le k-voisinge un clusterhead à qui s'affilier. Il diffuse alors un message CH dans son k-voisinage tout en indiquant son poids. Chaque nœud qui reçoit un message CH compare le poids de son clusterhead avec le poids reçu dans ce message. Si le poids reçu est supérieur, il peut se joindre à ce nouveau clusterhead sous certaines conditions.

Nous avons introduit certains paramètres qui conditionnent les réaffiliations, à savoir k, Pmin, Dmin et Dmax. K représente le nombre maximal de sauts entre un clusterhead et ses membres. Pmin représente le poids minimal que peut avoir un nœud pour assumer correctement le rôle de clusterhead. Dmin est le nombre minimal de nœuds par cluster au dessous duquel le cluster est considéré instable et donc sujet à des réaffiliations. Finalement Dmax désigne le nombre maximal de nœuds par cluster, qui, une fois atteint, le clusterhead refuse les nouvelles demandes d'affiliations à son cluster.

Notons qu'une réaffiliation peut se produire lorsque :

– un nœud membre se déplace d'un cluster à un autre,

– un clusterhead devient membre,

– un nœud membre devient un clusterhead.

Les réaffiliations peuvent uniquement survenir suite à la réception d'un message CH ou suite à la perte de connexion avec son clusterhead. Chaque nœud détient les informations suivantes : identificateur et poids de son clusterhead, densité de son cluster (nombre de nœuds par cluster) et le nombre de sauts à son clusterhead.

Un nœud peut joindre un cluster voisin si :

– Le clusterhead se trouve à au plus k sauts de lui ;

– La densité du cluster voisin n'a pas atteint la densité maximale Dmax.

Un nœud qui reçoit un message CH envoyé par un clusterhead de poids meilleur que celui de son clusterhead actuel, ne se réaffilit au nouveau clusterhead que si :

– La densité de son cluster est inférieure à Dmin,

– La densité de son cluster est supérieure à Dmin mais son poids est inférieur à Pmin.

L'idée ici étant qu'il n'est pas nécessaire pour les nœuds de s'affilier toujours au meilleur clusterhead tant que les performances de leur clusterhead courant sont « accpetables » et ceci afin d'améliorer la stabilité des clusters et réduire les mouvements des nœuds entre les clusters.

Un nouveau nœud qui joint le réseau est sommé de :

– Attendre une certaine période pendant laquelle il écoute son voisinage dans la recherche d'un cluster qu'il peut joindre.

– S'il trouve un cluster voisin qui vérifie les conditions citées plus haut, il le joint et notifie le clusterhead concerné par un message Join.

– S'il lui est impossible de joindre un cluster voisin, il prend la décision d'être clusterhead et informe les nœuds se trouvant dans son k-voisinage par un message CH.

Ainsi, on ne définit pas une phase de clustering proprement dite. Les nœuds sont élus clusterhead à chaque fois que cela est nécessaire. De cette manière, on s'affranchit de l'hypothèse, peu réaliste, impliquant que les nœuds ne bougent pas pendant la phase de clustering.

Une fois les clusters formés, il y a un échange périodique de messages Keep-Alive entre les clusterhead et leurs membres. Ces messages permettent d'une part de s'assurer qu'ils font encore partie du même cluster et d'autre part d'échanger des informations utiles dans le cas où un nœud est amené à changer de cluster.

4. ETUDE DE PERFORMANCES

Nous avons mené une série de simulations afin d'évaluer les performances du mécanisme de clustering proposé. Nous avons utilisé pour cela le simulateur NS2, dans lequel nous avons implémenté l'algorithme de clustering DRC. L'utilitaire « setdest » de NS2 a été utilisé pour généré les scénarios de mobilité des nœuds selon le modèle de mobilité « Random Waypoint ». Dans notre cas, nous avons fait varier la vitesse des nœuds en maintenant les temps de pause constants.

Pendant les simulations, nous nous sommes intéressés aux métriques suivantes :

- Le taux de réaffiliations : changement d'un nœud d'un cluster à un autre.

- Le taux d'élections : déclaration d'un nœud comme clusterhead

- La durée de vie moyenne des clusterhead

- Le nombre moyen de clusters.

Nous avons varié les vitesses des nœuds (de 0.2 à 10 m/s). L'aire de simulation considérée est de 700*500, avec une portée de 150m. Le nombre de nœud est de 70. Nous avons considéré trois valeurs de k (rayon maximum des clusters) à savoir 1, 2 et 3. Le temps de pause est de 20s. La densité minimale Dmin a été fixée à 3 et la densité maximale fixée à 9. Les poids ont été calculés en considérant la vitesse de nœuds, l'énergie disponible et la connectivité. Un poids minimal a été fixé pour chaque scénario. Les simulations ont une durée de 3000s et des valeurs moyennes sont calculées sur des blocs de 300s.

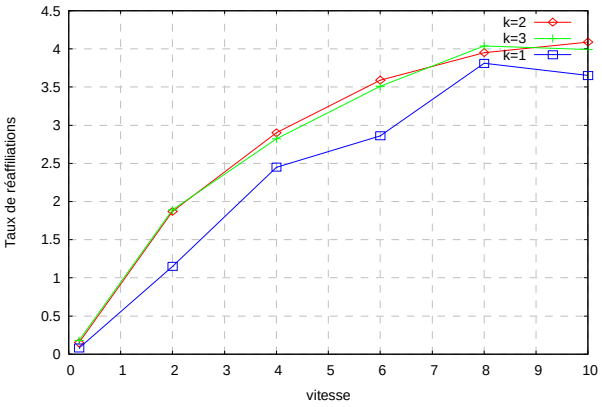

Figure 4.1 : Taux de réaffiliations en fonction de la vitesse

Les Figures 4.1 et 4.2 représentent respectivement les taux de réaffiliations et d'élections en fonction de la vitesse. Ces deux taux augmentent en fonction de la vitesse des nœuds. En effet, plus la vitesse est grande, plus la probabilité qu'un nœud se trouve hors de son cluster suite à un mouvement est grande. En revanche, nous remarquons que pour cette configuration, les réaffiliations deviennent presque constantes à partir de la vitesse 8 m/s. On remarque aussi que les valeurs de k (1, 2 et 3) influent peu sur les réaffiliations et les élections ce qui laisse plus de flexibilité à l'utilisateur pour le choix du rayon des clusters.

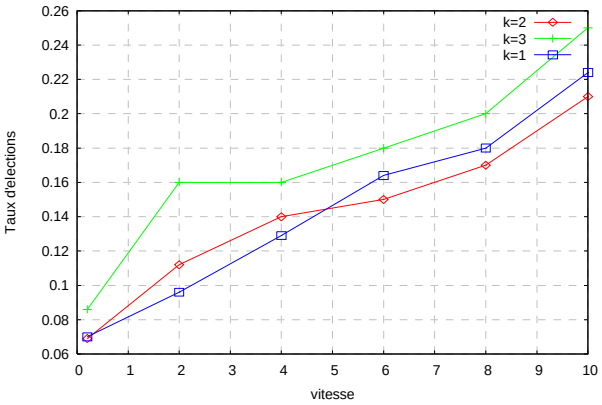

Figure 4.2 : Taux d'élections en fonction de la vitesse

Les durées de vie représentées dans la Figure 4.3 diminuent en fonction de la mobilité puisque la mobilité des nœuds introduit plus d'instabilité dans le réseau. Toutefois, on remarque que cette diminution n'est pas trop importante surtout pour les valeurs de k =1 et k =2.

La Figure 4.4 représente le nombre moyen de clusters générés. On remarque que ce nombre diminue faiblement en fonction de la vitesse. Ce résultat montre qu'en cas de mobilité, les nœuds sont capables de se réorganiser et de joindre des clusters déjà existants.

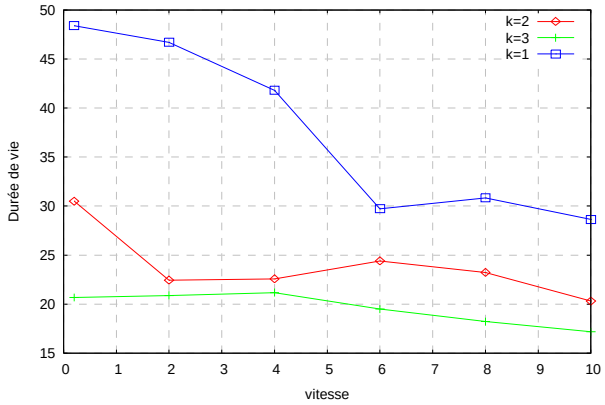

Figure 4.3 : durée de vie des clusterhead en fonction de la vitesse

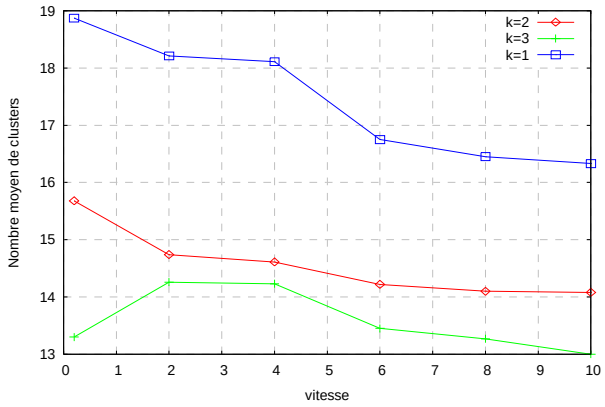

Figure 4.4 : nombre moyen de clusters en fonction de la vitesse

5. CONCLUSION

Dans ce papier, nous avons présenté un nouvel algorithme de clustering adéquat à la gestion par politiques dans les réseaux ad hoc. Il permet de construire des clusters de taille et de rayon configurables, dont la structure évolue en fonction des changements de topologie induits par la mobilité des nœuds. L'originalité de l'algorithme réside dans la gestion des réaffiliations des nœuds entre clusters. Pour cela des facteurs de contrôle ont été introduits afin de réduire les réaffiliations et améliorer la stabilité. Bien qu'il soit proposé pour la gestion par politiques, notre algorithme est assez générique et peut être utilisé pour d'autres objectifs.

En perspective, nous envisageons de faire une étude plus approfondie des performances de notre algorithme en étudiant l'influence des différents paramètres (Pmin, Dmin, Dmax) et en le comparant aux solutions existantes.

6. REFERENCES

[1] A. Amis, R.Prakash, T.Vuong, D. Huong, « Max-Min D-cluster formation in wireless ad hoc networks », in Proc. of IEEE Infocom, Tel Aviv, Mars 2000.

[2] M. Ayari M. and F. Kamoun, "Towards a Policy-Based Management For Adhoc Networks", The Third Annual Mediterranean Ad Hoc Networking Workshop (MedHoc 04), June 2004.

[3] D.J Baker and A. Ephremides, "The architetural organization of a Mobile Radio Network via a distributed algorithm", IEEE Transactions on Communications, COM-29, 1981.

[4] P. Basu, N.Khan, D Thomas and C. Little, "A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks", 21st International Conference on Distributed Computing Systems Workshops (ICDCSW '01), 2001.

[5] S. Basagni, "Distributed and mobility-adaptive clustering for multimedia support in multi-hop wireless networks", Proc. of IEEE VTS 50th Vehicular Technology Conference, 1999.

[6] H. Chaouchi and G. Pujolle, "Qos, security and mobility management for fixed and wireless networks under policy based techniques", IFIP World Computer Congress, 2002.

[7] G. Chen, F. Nocetti, J.S. Gonzalez, and I. Stojmenovic, "Connectivity-based K_hop clustering in wireless networks", in Proc. of the 35th Hawaii International Conference on System Sciences (HICSS-35), Janvier 2002.

[8] M. Chatterjee, K. Das and D. Turgut, "WCA: A Weighted Clustering Algorithm for Mobile Ad Hoc Networks," Journal of Cluster Computing, No. 5, 2002.

[9] S. Corson and J. Macker, "Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations", RFC 2501, IETF, January 1999.

[10] M. Gerla, and J. Tsai, "Multicluster, mobile, multimedia radio network", ACM-Baltzer Journal of Wireless Networks, Vol.1, No.3, pp.255-265, 1995.

[11] S. Ghiasi, A. Srivastava, X. Yang and M. Sarrafzadeh, "Optimal Energy Aware Clustering in Sensor Networks," Sensors Magazine, 2002.

[12] T.J Kwon and M. Gerla, "Effecient flooding with passive clustering in ad hoc networks", ACM SIGCOMM Computer Communication Review, January 2002.

[13] C. Lin and M. Gerla, "Adaptive Clustering for Mobile Wireless Networks", IEEE Journal on Selected Areas in Communications, Vol. 15, No. 7, September 1997.

[14] K. Phanse, L. DaSilva and S. Midkiff, "Extending Policy-Based Management to Ad Hoc Networks", Virginia Polytechnic Institute and State University, IREAN Research Workshop-2003, Alexandria, April 2003.

[15] A. Siddiqui, R. Prakash, "Effect of availability factor threshold and clustering gap on performance of clustering mechanisms for multi-cluster mobile ad hoc networks", IEEE International Conference on Communications, ICC 2002.

[16] S. Sivavakeesar, G. Pavlou, C. Bohoris and A. Liotta, "Effective Management through Prediction-Based-Clustering Approach in the Next-Generation Ad hoc Networks", In the Proc. of the IEEE International Conference on Communications (ICC '04), France, June 2004.