

---

# Étude de la formation de convois dans un réseau de véhicules sur autoroute

Florent Kaiser <sup>\*</sup>, Colette Johnen <sup>\*\*</sup>, Véronique Vèque <sup>\*\*\*</sup>

<sup>\*</sup> IFSTTAR - LEOST

florent.kaiser@ifsttar.fr

<sup>\*\*</sup> Université Bordeaux 1 - LABRI

johnen@labri.fr

<sup>\*\*\*</sup> Université Paris-Sud - Laboratoire des Signaux et Systèmes

veronique.veque@u-psud.fr

---

*RÉSUMÉ. Les réseaux de véhicules tendent désormais vers une architecture cellulaire hybride étendue avec une partie ad hoc. Les communications sont donc soit entre véhicules, soit du véhicule vers l'infrastructure. Les avantages de cette architecture sont multiples : meilleure connectivité, augmentation du débit et passage à l'échelle cependant du fait que la vitesse relative entre un nœud mobile et un point d'accès est importante, un tel réseau de véhicules est soumis à de fréquents handovers des véhicules d'un point d'accès à un autre. Dans cette étude, nous développons un protocole de formation de convois. Nos résultats montrent que notre approche est intéressante en termes de stabilité du convoi et de longueur du convoi.*

*ABSTRACT. The vehicular Ad Hoc network (VANET) is often considered as an ad hoc extension of a cellular mobile network. Thus communications are then possible between vehicles or from vehicle to infrastructure based-network. Major improvements take advantage of this architecture: connectivity, higher bandwidth and scalability. However, the relative speed difference between a mobile node and the access point triggers numerous vehicle handovers. In this paper, we introduce and develop a new protocol of clustering in vehicular networks. It consist to organize the communication into intrinsic convoy of vehicles. Finally, we show within simulation experiments that our proposal allows to build stable convoys with limited size.*

*MOTS-CLÉS : réseau véhicules, réseau ad hoc hybride, hiérarchisation, stabilité*

*KEY WORDS: vehicular network, hybrid ad hoc network, clustering protocol, stability*

---

## 1. Introduction

Les réseaux *ad hoc* de véhicules ou *VANET* apparaissent depuis quelques années comme un moyen effectif d'améliorer la sécurité routière. Les véhicules sont équipés de moyens de communication radio qui leur permettent de communiquer entre véhicules (*Vehicle-to-Vehicle*, V2V). Si un véhicule détecte une anomalie sur la route, par exemple la présence de verglas détectée par des capteurs, il transmet un message d'alarme vers les véhicules voisins à portée radio. Les véhicules récepteurs analysent et évaluent le message et peuvent prendre les mesures qui s'imposent c'est-à-dire ralentir. Dans cet exemple, l'échange de messages a permis de prévoir un problème avant que le véhicule ne soit en mesure de le détecter. On parle alors de vision augmentée. Le deuxième type d'application est dit *orientée utilisateur*. Il s'agit alors des applications classiques Internet ou à portée géographique par exemple, la publicité pour les restaurants d'une région donnée. Pour ce dernier cas, on considérera en plus des communications V2V les communications V2I (*Vehicle-to-Infrastructure*), qui permettent à un véhicule de se connecter au réseau mobile par une des stations de base installées le long de la route. Dans [KAI 09], nous avons étudié la faisabilité d'un réseau de véhicules basé sur une architecture hybride utilisant ces deux types de communication. Cette architecture hybride permet aux véhicules de communiquer entre eux en mode Ad hoc ou avec un point d'accès faisant partie d'une infrastructure de réseau fixe, connectée à Internet. Les avantages de cette architecture hybride sont multiples. Elle permet d'étendre la couverture des réseaux de type cellulaires (comm UMTS ou LTE) à moindre coût. Elle résout le problème du partitionnement des réseaux *ad hoc* de véhicules et permet le passage à l'échelle des réseaux étendus ou denses, tout en offrant un plus haut débit. Pour le passage à l'échelle, l'infrastructure fixe dispose d'une plus grande bande passante que la partie ad hoc, permettant à la fois un réseau plus long mais aussi avec plus de nœuds. Néanmoins, du fait que la vitesse relative entre un nœud mobile et un point d'accès est importante, un tel réseau de véhicules est soumis à de fréquents handovers des véhicules d'un point d'accès à un autre. La fréquence des handovers cause des pertes de paquets et limite le débit utile du réseau. C'est pourquoi, dans cette étude, nous nous sommes intéressés à une organisation des véhicules en "cluster" ou convois afin de reporter le handover au niveau du convoi plutôt qu'individuellement par nœud.

Ce papier est organisé comme suit : tout d'abord, nous résumons les travaux antérieurs concernant la formation de *cluster* en général, puis appliquée aux réseaux *ad hoc*. Ensuite, nous présentons notre protocole de formation de convois. Enfin, nous proposons une évaluation de ce protocole à l'aide de simulations.

## 2. Travaux antérieurs

La hiérarchisation du réseau permet de l'organiser en groupes de nœuds ou *cluster* avec un routage différencié interne au *cluster* et entre les *clusters*. Une telle division de l'espace permet de réduire l'overhead de routage (diffusion de requêtes) et un meilleur passage à l'échelle.

Le premier sous-problème réside dans la manière de constituer un groupe ou *cluster* et est réalisé par l'algorithme de formation de *clusters*. Dans le groupe de nœuds constitué, certains nœuds ont un rôle particulier notamment de *clusterhead*, de passerelle pour interconnecter les groupes ou de collecteur d'informations. Les algorithmes de formation de *clusters* peuvent être classés selon leur

manière de contrôler la taille d'un *cluster* définie soit par le nombre maximum de sauts entre chaque membre et son *clusterhead*, soit par le nombre maximum de nœuds dans le *cluster* ou par le nombre maximum de *clusters* dans le réseau. Le *clusterhead* est déterminé par un algorithme d'élection qui peut constituer l'initialisation du *cluster*. L'arrivée, le départ des nœuds dans le *cluster* et le changement de *clusterhead* sont faits dans la phase de maintenance grâce aux messages CLUSTERHEAD et JOIN.

Les algorithmes DCA (Distributed Clustering Algorithm) et DMAC (Distributed Mobility-Adaptive Clustering)[BAS 99] sont les plus souvent cités et permettent de constituer des *clusters* 1-saut où chaque membre du cluster a au moins un *clusterhead* comme voisin, le *clusterhead* étant le voisin de plus gros poids. Le poids peut représenter le degré d'un nœud, l'inverse de sa vitesse moyenne, un identifiant, etc. DMAC est mieux adapté que DCA aux réseaux dynamiques, car il introduit un algorithme de maintenance du *cluster* qui réagit aux changements de topologies de niveau MAC : nouveau lien et rupture d'un lien. De plus, il autorise un plus grand nombre de *clusters* assurant ainsi une plus grande stabilité de la structure hiérarchique.

Pour minimiser le nombre de *clusters*, les auteurs de [NOC 03] ont généralisé l'algorithme 1-saut à k-sauts. Pour découvrir le voisinage à k-sauts, les messages CLUSTERHEAD et JOIN sont diffusés avec un TTL égal à k. Il est également possible de limiter la taille d'un *cluster*, non pas par le nombre de sauts maximum entre un membre et son *clusterhead*, mais par le nombre maximum de nœuds dans un *cluster*. Dans [OHT 03], si le *cluster* dépasse la taille limite, il est divisé en deux (par élection d'un nouveau *clusterhead*). Si le *cluster* est considéré comme trop petit, un nœud en bordure d'un *cluster* voisin plus grand est recruté. Enfin, le *cluster* peut être constitué à partir d'un arbre couvrant minimal sur le graphe du réseau *ad hoc* [NIK 00].

La formation de *clusters* dans un réseau *ad hoc* n'est pas une fin en soi mais un moyen d'améliorer le routage *ad hoc* en limitant ses temps de recherche ou l'overhead de signalisation. Ainsi, [NIU 06] propose un protocole de routage *ad hoc* basé sur un algorithme de formation de *clusters* k-sauts. Le réseau *ad hoc* est hiérarchisé en deux niveaux : le niveau supérieur (niveau des *clusters*) et le niveau inférieur (niveau des nœuds). Au niveau du *cluster*, le routage est proactif : chaque *clusterhead* maintient un graphe de l'ensemble des clusterheads du réseau et une liste des membres du *cluster* associé. Une autre stratégie pour améliorer les performances d'un routage réactif est proposé dans [OHT 07] avec formation des *clusters* selon [OHT 03]. Un *clusterhead* construit un arbre de recouvrement minimal à partir des informations sur la topologie de son *cluster* obtenu à l'aide de messages envoyés périodiquement par les membres de son *cluster*. Les requêtes de routage sont diffusées via les nœuds de l'arbre, et non à tous les nœuds du réseau, réduisant ainsi l'overhead. Pour améliorer le passage à l'échelle du protocole de routage proactif OLSR, les auteurs de [ROS 07] considèrent la formation de *clusters* sans détailler l'algorithme utilisé.

Dans les réseaux *ad hoc* de véhicules, les techniques de formation de *clusters* peuvent pallier un grave inconvénient des protocoles de routage basés sur la position géographique [BAH 00, KAR 00]. En effet, le point faible de ces protocoles, d'un point de vue du passage à l'échelle, est la localisation géographique de la destination qui est réalisée de manière gloutonne [DAS 05]. Dans [RAÚ 06], les auteurs proposent une solution pour améliorer le passage à l'échelle d'un protocole de localisation géographique à l'aide de *clusters* dans un réseau de véhicules.

D'autres travaux proposent des algorithmes de formation de *clusters* adaptés aux réseaux de véhicules et traitent de leur évaluation. Dans [FAN 05], les auteurs prennent en compte la vitesse ou la position des véhicules voisins dans l'algorithme d'élection de *clusterhead* (k-sauts Lowest-ID et Highest-Degree), afin d'augmenter la stabilité du *cluster*. Dans [FAN 07], ces travaux sont étendus pour prendre en compte la direction de déplacement des véhicules : deux véhicules circulant dans la direction opposée ne peuvent pas être dans le même *cluster*. Les auteurs ajoutent également un critère pour prendre en priorité un *clusterhead* qui a été *clusterhead* le plus longtemps dans le passé.

Il est également possible de prendre en compte la dynamique particulière d'un réseau de véhicules, par exemple il existe un effet oscillatoire de la distance entre les véhicules. Dans [BLU 03], cet effet est pris en compte avec un critère de distance (géographique) maximale entre deux *clusterheads*. Ils ajoutent également un critère basé sur le point de sortie du véhicule de la route.

Les différents travaux que nous avons présentés donnent un aperçu de la complexité du problème et aussi de son ancienneté. Notre proposition est originale car elle tient compte de la dynamique particulière des véhicules circulant sur une autoroute. En effet, il se forme naturellement des regroupements de véhicules que nous appelons convois. Notre approche se base sur ces convois naturels pour les organiser et permettre simplement les communications internes au convoi. Dans la suite, nous présentons notre protocole de formation de *clusters* adapté aux réseaux de véhicules.

### 3. Formation de convois

Nous avons considéré un réseau de véhicules sur autoroute, où les véhicules se suivent sur une ligne. De plus, nous avons montré dans [MAB 06] que la dynamique particulière des véhicules dans un tel réseau *ad hoc* entraînait la formation de convois. Dans un convoi, les véhicules se suivent, le *clusterhead* est alors tout désigné : c'est le véhicule en tête du convoi. Le système de géolocalisation permet à la fois de connaître la vitesse, la position et la direction de déplacement du véhicule. Ces trois informations sont nécessaires à notre algorithme de formation de convois pour déterminer si un nœud peut rejoindre le convoi, afin d'en assurer la stabilité. Dans cette partie, nous proposons notre algorithme réparti de formation de convoi. Nous considérons que le véhicule est équipé d'une seule interface de transmission sans fil, d'un système de géolocalisation, d'un système de mesure de la vitesse actuelle.

Nous souhaitons former des *clusters* de véhicules stables à k-sauts pouvant communiquer entre eux via un protocole de routage *intra-cluster*. À l'intérieur du *cluster*, le graphe représentant le réseau doit alors être connexe. Nous ne prenons pas en considération la voie où le véhicule circule, la précision des systèmes de géolocalisation étant d'une dizaine de mètres. Néanmoins, le regroupement de véhicules sur une même voie est favorisé par le critère d'entrée dans un convoi, que nous verrons ci-après. Comme montré sur la Figure 1, les véhicules qui circulent dans une direction opposée ne sont pas dans le même convoi. En revanche dans le convoi 7, les véhicules sont dans le même convoi alors qu'ils sont sur des files différentes. Les vitesses des véhicules des convois 1 et 4 sont très différentes ce qui explique qu'ils se trouvent dans deux convois différents.

Dans un but de simplicité, nous choisissons de séparer le protocole de formation de convoi en lui-même du protocole permettant la communication entre les nœuds.

### 3.1. Protocole de routage intra-convoi

A l'intérieur d'un convoi, le protocole de routage est le plus simple possible et s'inspire de [DUC 07]. Il se base sur la position géographique des véhicules et la direction du convoi. La diffusion d'un paquet dans la direction (respectivement dans la direction opposée) du déplacement du convoi permet de délivrer un message à tous les véhicules du convoi en amont (respectivement en aval) du véhicule initiateur du message. Le protocole de formation de convoi n'a donc pas la charge de la transmission des messages.

### 3.2. Algorithmes de formation de convois

Notre algorithme de formation de convois reprend les principes des algorithmes de formation de *clusters*. Le chef de convoi, ou *clusterhead*, est tout simplement le nœud en *tête de convoi*, c'est-à-dire le nœud devant tous les véhicules membres du convoi (Figure 1). Les algorithmes de formation de *clusters* que nous avons décrits limitent la taille du *cluster* soit par le nombre de sauts maximum jusqu'au *clusterhead* ou par le nombre de nœuds membres du *cluster*. Notre algorithme définit la taille maximale du *cluster* par sa longueur géographique (en mètres), c'est-à-dire la distance maximale séparant la tête de convoi de la queue de convoi (Figure 1).

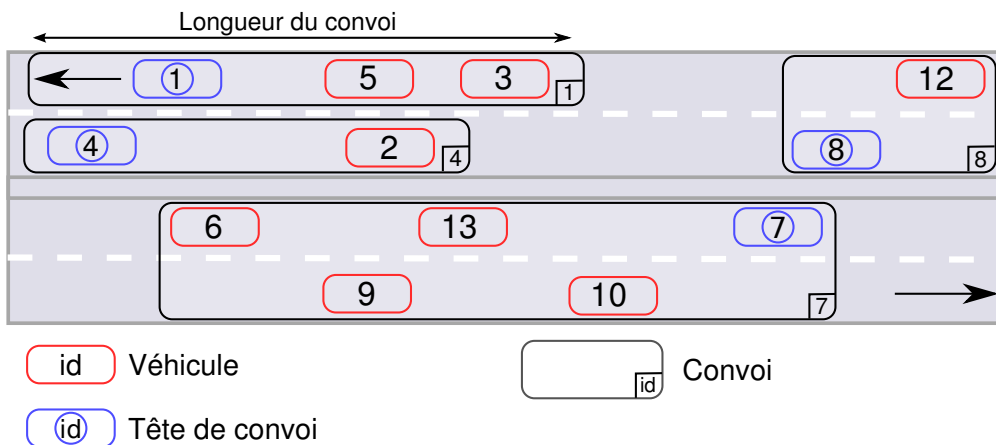


Figure 1 – Convois de véhicules sur une autoroute 2x2 voies. La tête de convoi est le nœud devant tous les véhicules du convoi.

Initialement, tout véhicule est seul dans son propre convoi et donc tête de convoi et il existe donc autant de convois qu'il y a de véhicules dans le réseau. Un nœud déclare sa présence à ses voisins en envoyant à intervalles réguliers un message HELLO, contenant son identifiant, l'identifiant de son convoi, sa position et sa vitesse.

La formation et la maintenance de convoi sont régencées par deux processus distincts : la *fusion* et la *scission* de convois. La fusion permet à un convoi de s'agrandir en recrutant de nouveaux véhicules. La scission est provoquée quand le convoi n'est plus connexe : il existe un nœud ne

Nom du message	Sigle	Mode de diffusion
Join convoy request	JREQ	Direction du déplacement
Join convoy reply	JREP	Vers une position
Node join	NJ	Vers une position
Convoi info	CI	Direction opposée au déplacement
Acquittement d'un CI	CIACK	Direction du déplacement
Hello	HELLO	Voisinage

Tableau 1 – Liste des messages utilisés par les algorithmes de formation de convoi.

pouvant plus communiquer avec la tête de convoi. La gestion de la scission de convoi permet la maintenance du *cluster*. L'ensemble des messages utilisés pour la formation et la maintenance des convois est résumé dans le Tableau 1.

### 3.2.1. Fusion de convois

Dans la suite, nous allons décrire les algorithmes déclenchés lors de la réception d'un message.

Pour pouvoir initier la fusion de convois  $C_1$  et  $C_2$ , le *clusterhead* de  $C_1$  (ou  $C_2$ ) doit être le *voisin* (*clusterhead* ou non) d'un nœud contenu dans le *cluster*  $C_2$  (ou  $C_1$ ). À partir des informations contenues dans le message HELLO envoyé par le voisin, le *clusterhead* décide de se porter ou non candidat à la fusion.

Les conditions de candidature sont :

- La durée de vie du lien entre le *clusterhead* et le voisin est supérieure à la limite *LifeTimeLimit* fixée par le protocole, ce qui permet de garantir une durée de vie du *cluster* supérieure à cette limite (si nous considérons la vitesse des nœuds membres constante)
- Les deux convois  $C_1$  et  $C_2$  se déplacent dans la même direction.
- $C_1$  et  $C_2$  ont des identifiants différents, sinon la fusion n'a pas d'intérêt.
- Le nœud voisin est devant le *clusterhead*,
- Une candidature de fusion entre  $C_1$  et  $C_2$  n'a pas déjà eu lieu récemment.

Après fusion, le *clusterhead* étant toujours devant tous les nœuds membres du *cluster*, le *clusterhead* candidat devient membre du *cluster* voisin.

Si toutes ces conditions sont vérifiées, le *clusterhead* candidat envoie un message *Join convoy request* (JCREQ) au nœud voisin qui le transmet à son *clusterhead*, que nous appelons *clusterhead* *voisin*. Le *clusterhead* *voisin* vérifie la taille des deux convois réunis, si elle est inférieure à la taille maximale de convois, alors il envoie un message *Join convoy reply* (JCREP) au *clusterhead* candidat confirmant la fusion. Ce message est envoyé vers la position géographique de la destination. Si la taille dépasse la taille maximale, alors le *clusterhead* *voisin* ne fait rien et le processus de fusion est avorté. Sur réception du message JCREP, le *clusterhead* candidat envoie un message *Node join* (NJ) au *clusterhead* *voisin* contenant la liste des nœuds de son *cluster*, diffuse un message *Convoi info* (CI) dans la direction opposée au déplacement du *cluster* et met à jour l'identifiant du convoi auquel

il appartient. Sur réception du message NJ, le *clusterhead* ajoute les nœuds contenus dans le message dans sa table de nœuds du *cluster*. Sur réception d'un message CI, un membre du *cluster* met à jour l'identifiant de *cluster* avec celui du *clusterhead* voisin.

### 3.2.2. Scission de convois

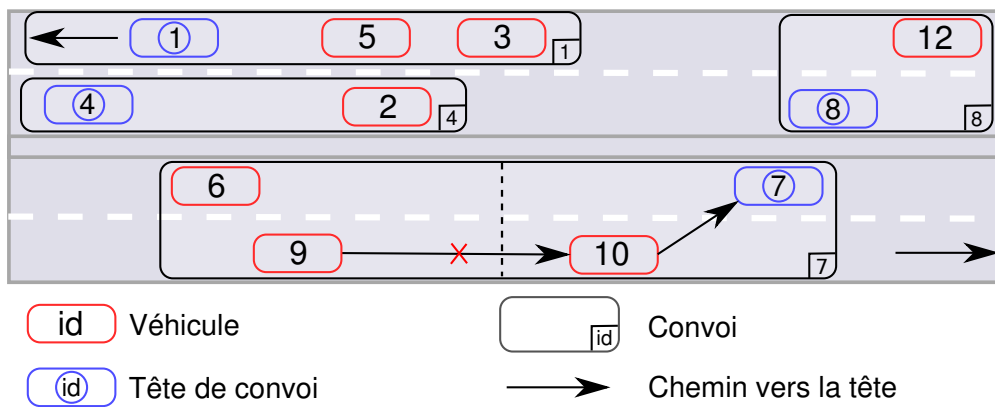


Figure 2 – Scission du convoi 7. Après scission, le convoi 7 subsiste et un nouveau convoi 9 s'est formé.

La gestion de la scission de convois joue le rôle de maintenance du *cluster*. La détection de rupture d'un lien avec un membre voisin ne suffit pas à déduire que le nœud quitte le *cluster*. En effet, il peut encore communiquer avec tous les membres du *cluster* et le *clusterhead* par l'intermédiaire d'un autre lien. Par définition, la scission d'un convoi se produit quand un nœud membre ne peut plus communiquer avec le *clusterhead*, c'est-à-dire qu'il n'existe pas de chemin vers le *clusterhead* (Figure 2). Le réseau étant linéaire, aucun des nœuds en aval du membre ne peut non plus communiquer avec le *clusterhead*. Le membre en tête devient alors *clusterhead* et l'ensemble des nœuds en aval appartenant à l'ancien *cluster* rejoignent le nouveau *cluster*.

Le *clusterhead* diffuse à intervalles réguliers, dans la direction opposée au déplacement de *cluster*, un message *Convoi info* (CI) à tous les membres du *cluster*. Lorsqu'un membre ne reçoit pas ce message depuis un délai prédéterminé, il considère qu'il n'existe plus de chemin avec le *clusterhead*, le *cluster* n'est plus connexe et il y a scission du convoi. Il s'auto-proclame *clusterhead* et diffuse un message CI dans la direction opposée au déplacement du *cluster*, permettant aux nœuds en aval de changer de *cluster*.

Sur réception d'un message CI, un membre du *cluster* affecte son identifiant de *cluster* au nouvel identifiant du *clusterhead* et acquitte la réception du message à son nouveau *clusterhead*. Sur réception de l'acquiescement, le nouveau *clusterhead* ajoute le membre du *cluster* dans sa table. Le *clusterhead* en amont, ne recevant pas d'acquiescement au message CI, considère que ses membres ne font plus partie de son *cluster* et les enlève de sa table. Le processus de scission est alors terminé.

#### 4. Évaluation du protocole de formation de convois

Nous avons mis en œuvre l’algorithme de formation de convois dans le simulateur JIST/SWANS [JiS] au niveau application. En effet, le routage des messages envoyés et diffusés par le protocole de formation de convois est délégué au protocole de routage géographique. Afin de simuler finement le déplacement des véhicules sur une autoroute, nous utilisons le simulateur de trafic routier que nous avons développé [MAB 07] pour modéliser la mobilité des nœuds dans un contexte autoroutier. En plus des paramètres liés au trafic routier (longueur de l’autoroute et densité de véhicules), nous utilisons quatre paramètres liés à la simulation de formation de convoi qui sont donnés dans le Tableau 2. Dans la suite, si le paramètre ne varie pas, nous utilisons cette valeur par défaut.

Temps de simulation	600 s
Intervalle des Hello	2 s
Durée de vie limite d’un lien du convoi	600 s
Longueur limite d’un convoi	2 km
Longueur de l’autoroute	10 km
Densité de véhicules	10 véhicules/voie/km
Portée radio	250 m
Bande passante	11 Mb/s

Tableau 2 – Paramètres de simulation par défaut

Nous nous intéressons à cinq métriques : le nombre de ruptures de convois, le nombre de convois, le taux de partitionnement, la longueur et la taille d’un convoi. *Le nombre de ruptures de convois* pendant une simulation permet d’évaluer la *stabilité* globale des convois formés : plus le nombre de ruptures est important, moins les convois sont stables. *Le nombre de convois* permet d’évaluer la *qualité* de leur formation : au pire il existe autant de convois que de nœuds. Chaque convoi étant alors composé d’un seul nœud, la formation de convoi n’a alors aucun intérêt. Moins il y a de convois, plus les convois sont importants et plus la mobilité des véhicules peut entraîner des scissions de convois. Nous nous intéressons également au *taux de partitionnement* défini comme le nombre de convois divisé par le nombre de nœuds. Le taux de partitionnement est alors l’inverse de la taille moyenne des convois. Nous différencions *longueur d’un convoi* et *taille d’un convoi*. La longueur d’un convoi est la distance en km entre la tête et la queue (véhicule derrière tous les membres du convoi) de convoi. La taille d’un convoi est le nombre de véhicules membres du convoi.

Dans un premier temps, nous cherchons les paramètres optimaux de l’algorithme de formation de convoi. Dans un second temps, nous analysons la stabilité et la qualité des convois formés avec les paramètres optimaux trouvés.

##### 4.1. Paramètres optimaux de l’algorithme

La connectivité d’un convoi est proportionnelle à sa densité. Plus un convoi est dense, plus la taille d’un convoi est potentiellement grande, diminuant le taux de partitionnement. Cette hypothèse est vérifiée sur les résultats en Figure 3(a) : le taux de partitionnement diminue avec la densité.



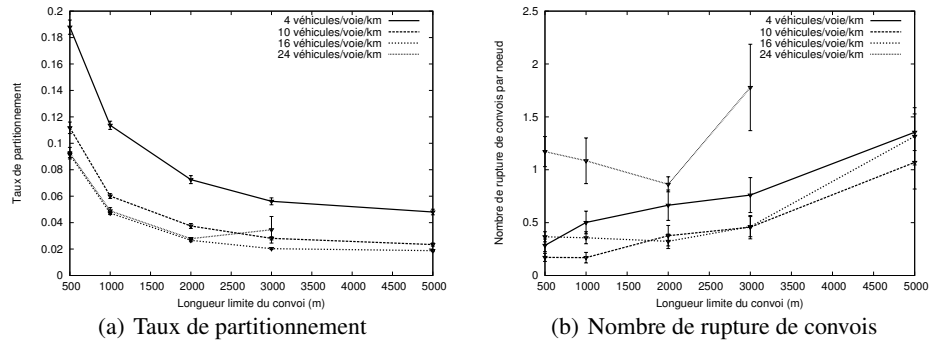


Figure 3 – Évaluation en fonction de la longueur limite de convoi pour plusieurs densités

Dans cette même figure, on voit que le taux de partitionnement est également proportionnel à la longueur limite du convoi. Par conséquent, le paramètre de longueur maximale des convois permet bien de limiter la taille des convois. La variation du taux de partitionnement est plus faible dès que la longueur limite est supérieure à 2000 m et quasi-constante à partir de 3000 m. Le nombre de ruptures de convois augmente avec la densité et la longueur limite des convois (Figure 3(b)). Pour une densité de 24 véhicules/km/voie, le nombre de ruptures explose à partir de 2000 m. Au delà de 3000 m et avec une densité de 24, la puissance requise par le simulateur est trop importante pour réaliser une simulation, d'où l'absence de mesures. Nous concluons qu'il n'est pas utile de fixer la longueur maximale d'un convoi au delà de 2000 m pour satisfaire le compromis d'un taux de partitionnement proche du minimum et d'un nombre de ruptures de convoi minimal. Nous avons donc fixé par défaut la limite de la longueur de convoi à 2000 m.

#### 4.2. Distribution de la longueur des convois

Nous avons discuté de la longueur maximale de convoi et nous souhaitons maintenant vérifier la distribution des longueurs de convoi. Cette distribution est calculée à la fin de la simulation. La Figure 4 présente cette distribution pour une limite fixée à 2000, 3000 et 5000 m. Chaque barre représente le nombre de convois compris entre une longueur  $l$  et  $l + 500$  m sur un échantillon de 100 convois environ réalisé en plusieurs simulations.

Nous remarquons deux anomalies apparentes : une proportion importante de convois de longueur comprise entre 0 et 500 m, et des convois de taille supérieure à la limite fixée. Quand un nœud arrive sur l'autoroute, il est tête de convoi et seul dans le convoi (aucun membre ne l'a encore rejoint). Il reste seul dans un temps limité avant de fusionner avec un autre convoi. Chaque nœud ayant été au départ dans un convoi de longueur zéro, la proportion de petits convois proches de la longueur zéro (un unique nœud dans le convoi) est naturellement importante, mais non significative. En effet un convoi reste peu de temps avec une taille inférieure à 10, comme nous le constatons sur la Figure 4. La seconde anomalie constatée est l'existence de convois de taille supérieure à la taille limite (2000 m). Les vitesses des véhicules étant proches les unes des autres, nous fixons la longueur limite de convoi seulement au moment de sa formation, c'est-à-dire lors de la fusion. En

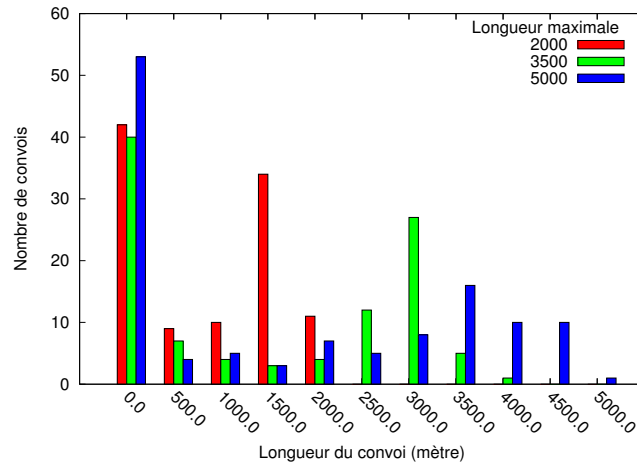


Figure 4 – Distribution des longueurs de convoi en metre

revanche, la topologie du convoi est dynamique : la distance relative entre les véhicules varie au cours du temps. Il est possible qu'un véhicule ayant une vitesse plus faible par rapport aux autres s'éloigne des autres véhicules du convoi et augmente la longueur du convoi au cours du temps. Par conséquent, bien qu'aucun nouveau véhicule n'ait rejoint le convoi, la longueur du convoi peut augmenter sans provoquer une rupture de convoi car la longueur reste proche de 2 km. La distribution est conforme aux attentes, notamment pour les convois de longueur 2000 et 3000 m, le maximum étant respectivement pour des convois de longueur comprise entre 1500 et 2000 m et 3000 et 3500 m. Pour une limite de 5000 m, la distribution est plus répartie, le maintien de convoi supérieur à 3000 m étant plus difficile à cause d'une probabilité rupture de convoi plus importante (Figure 3(b)).

#### 4.3. Stabilité des convois

L'objectif de la formation de convois est d'obtenir le moins de convois possibles ce qui implique des convois de grande taille. En revanche le nombre de ruptures de convoi est fonction de la taille des convois. Nous avons trouvé un compromis avec un convoi limité à 2000 m, mais nous ne présentons pas ce résultat ici. Nous nous intéressons donc à la durée d'un convoi ayant une taille  $n$ , avec une longueur limite de 2000 m. Sur la Figure 5, nous mesurons la distribution de la durée de vie d'un convoi en fonction de sa taille. Ici la durée de vie est exprimée en pourcentage de simulation, c'est-à-dire la part de temps pendant laquelle le convoi a une taille comprise dans l'intervalle. La répartition de la taille de convoi dépend de la densité, en effet avec une densité de véhicules à 4 véhicules par kilomètre et par voie, la taille est répartie entre 0 et 40, pour une densité de 10 entre 0 et 80, pour une densité de 16 entre 0 et 140. Nous remarquons également que les convois de grande taille sont ceux qui durent le plus longtemps, surtout avec des densités de 4 et 10 véhicules par kilomètre et par voie. Cette propriété est très intéressante pour obtenir des convois stables ayant une taille suffisamment grande pour justifier de la formation de convois dans un réseau de véhicules.

D'autres résultats sont exposés dans le manuscrit de thèse [KAI 10].

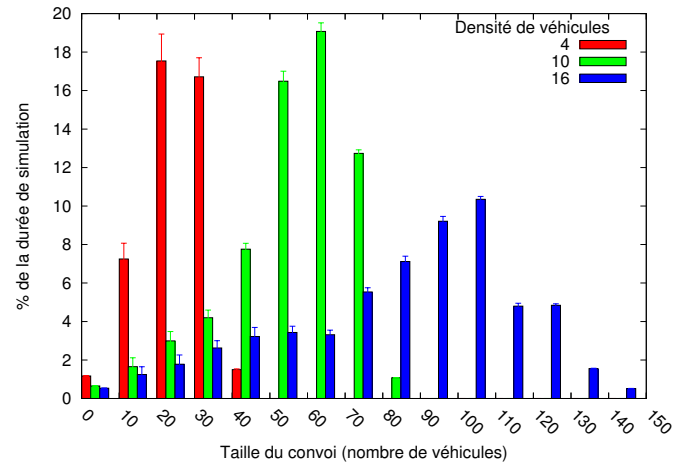


Figure 5 – Distribution de la durée d’un convoi en fonction de sa taille

## 5. Conclusion

Nous avons présenté un algorithme de formation de *clusters* original adapté au réseau de véhicules. Celui-ci repose sur la dynamique particulière des véhicules sur autoroute qui se regroupent naturellement pour former des convois. Notre algorithme de formation de convois consiste donc à désigner le *clusterhead* (la tête du convoi) et à définir une longueur maximale. Nous avons proposé deux algorithmes de fusion et de scission de convois. Le convoi est formé en prenant en compte la qualité des liens entre les nœuds et sa longueur maximale afin de former des convois aussi stables que possible. Les simulations réalisées à l’aide d’un simulateur de réseaux de véhicules dans un contexte autoroutier nous ont permis d’étudier les distributions de la taille et de la longueur des convois satisfaisantes. Ils ont montré la bonne stabilité des convois pour des convois longs de 2 km car il y a peu de ruptures de convoi étant donné qu’au sein d’un convoi les véhicules ont une vitesse relative faible.

Ces résultats nous permettent de penser qu’il est possible d’exploiter les propriétés de stabilité relative entre les nœuds d’un convoi afin de gérer plus finement la mobilité. Notre objectif est que seul le *clusterhead* exécute le handover avec une station de base ainsi que l’attachement à celle-ci.

## 6. Bibliographie

- [BAH 00] BAHL P., PADMANABHAN V., « RADAR : An in-building RF-based user location and tracking system », *IEEE infocom*, vol. 2, INSTITUTE OF ELECTRICAL ENGINEERS INC (IEEE), 2000, p. 775–784.
- [BAS 99] BASAGNI S., « Distributed clustering for ad hoc networks », *Parallel Architectures, Algorithms, and Networks, 1999. (I-SPAN '99) Proceedings. Fourth International Symposium on*, Perth/Fremantle, WA, Australia, 1999.
- [BLU 03] BLUM J., ESKANDARIAN A., HOFFMAN L., « Mobility management in IVC networks », *Intelligent Vehicles Symposium, 2003. Proceedings. IEEE*, juin 2003, p. 150–155.

- [DAS 05] DAS S., PUCHA H., HU Y., « Performance comparison of scalable location services for geographic ad hoc routing », *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies.*, vol. 2, 2005.
- [DUC 07] DUCOURTHIAL B., KHALED Y., SHAWKY M., « Conditional Transmissions : Performance Study of a New Communication Strategy in VANET », *Vehicular Technology, IEEE Transactions on*, vol. 56, n° 6, 2007, p. 3348 -3357.
- [FAN 05] FAN P., NELSON P. C., HARAN J., DILLENBURG J., « Cluster-Based Framework in Vehicular Ad-Hoc Networks », , 2005, p. 32–42, Springer-Verlag.
- [FAN 07] FAN P., MOHAMMADIAN A. K., NELSON P. C., HARAN J., DILLENBURG J., « A Novel Direction Based Clustering Algorithm in Vehicular Ad Hoc Networks », 86th Annual Transportation Research Board Meeting, January 2007.
- [JiS] « Jist / Swans : Java in Simulation Time / Scalable Wireless Ad hoc Network Simulator », <http://jist.ece.cornell.edu>.
- [KAI 09] KAISSE F., VÈQUE V., « On the Scalability Problem of Highway Ad hoc Network », IEEE, Ed., *Proceedings of WCNC'09 (Wireless Communications & Networking Conf.)*, April 2009.
- [KAI 10] KAISSE F., « Communications dans les réseaux fortement dynamiques », PhD thesis, juin 2010, <http://tel.archives-ouvertes.fr/tel-00512021/fr>.
- [KAR 00] KARP B., KUNG H. T., « GPSR : Greedy Perimeter Stateless Routing for Wireless Networks », *The sixth Annual ACM/IEEE International Conference On Mobile and Networking (MobiCom 2000)*, Aout 2000.
- [MAB 06] MABIALA M., BUSSON A., VÈQUE V., « Analyse du trafic et du routage dans un réseau Ad Hoc de véhicules », *Colloque Francophone sur l'Ingénierie des Protocoles - CFIP 2006*, 2006.
- [MAB 07] MABIALA M., BUSSON A., VÈQUE V., « Performance evaluation of VANET under realistic vehicular traffic assumption », *Traffic and Granular Flow (TGF'07)*, Springer Verlag, 2007.
- [NIK 00] NIKAEIN N., LABIOD H., BONNET C., « DDR-distributed dynamic routing algorithm for mobile ad hoc networks », *Mobile and Ad Hoc Networking and Computing, MobiHOC.*, Boston, MA, USA, 2000, p. 19–27.
- [NIU 06] NIU X., TAO Z., WU G., HUANG C., CUI L., « Hybrid Cluster Routing : An Efficient Routing Protocol for Mobile Ad Hoc Networks », *IEEE International Conference on Communications, ICC'06*, vol. 8, 2006.
- [NOC 03] NOCETTI F. G., GONZALEZ J. S., STOJIMENOVIC I., « Connectivity Based  $k$ -Hop Clustering in Wireless Networks », *Telecommunication Systems*, vol. 22, n° 1–4, 2003, p. 205–220.
- [OHT 03] OHTA T., INOUE S., KAKUDA Y., « An Adaptive Multihop Clustering Scheme for Highly Mobile Ad Hoc Networks », *Proceedings of the The Sixth International Symposium on Autonomous Decentralized Systems (ISADS'03)*, Washington, DC, USA, 2003, page 293.
- [OHT 07] OHTA T., MURAKAMI N., KAKUDA Y., « Performance Evaluation of Autonomous Clustering for Hierarchical Routing Protocols in Mobile Ad Hoc Networks », *ICDCSW '07*, Washington, DC, USA, 2007, page 56.
- [RAÚ 06] RAÚL AQUINO S., ARTHUR EDWARDS B., « A Reactive Location Routing Algorithm with Cluster-Based Flooding for Inter-Vehicle Communication », *Computación y Sistemas*, vol. 9, n° 4, 2006, p. 297–313.
- [ROS 07] ROS F. J., RUIZ P. M., « Cluster-based OLSR extensions to reduce control overhead in mobile ad hoc networks », *IWCMC '07 : Proceedings of the 2007 international conference on Wireless communications and mobile computing*, New York, NY, USA, 2007, ACM, p. 202–207.