

---

# Un Mécanisme de Synchronisation pour les Réseaux Sans Fil Multi-Sauts

**Bachar Wehbi, Anis Laouti, Wissam Mallouli et Ana Cavalli**

GET/INT CNRS/SAMOVAR  
9 rue Charles Fourier  
F-91011 Evry Cedex FRANCE  
*prénom.nom@int-evry.fr*

---

*RÉSUMÉ.* Avec le développement des technologies sans fil, les réseaux informatiques ont connu ces dernières années un essor sans précédent. De nouveaux types de réseaux ont fait leur apparition comme les réseaux sans fil multi-sauts. La synchronisation des nœuds dans de tels réseaux est primordiale pour des services collaboratifs ainsi que pour certains mécanismes de sécurité. Dans cet article, nous décrivons un protocole de synchronisation pour les réseaux multi-sauts basé sur un mécanisme de synchronisation de récepteur-à-récepteur. Ce mécanisme permet de définir une fonction d'ajustement entre les horloges locales de différents nœuds et un nœud de référence. L'avantage de ce protocole est, d'une part, l'élimination des sources d'indéterminisme affectant la synchronisation et, d'autre part, la synchronisation de tous les nœuds par rapport à un nœud de référence tout en minimisant la charge de communication.

*ABSTRACT.* In the recent years, the advances in wireless technologies have led to the deployment of new types of networks such as the auto-organized wireless multihop networks. The node synchronization in such networks is of high importance for cooperative services and some security mechanisms. In this paper, we describe a receiver-to-receiver based synchronization mechanism for wireless multi-hop networks. It defines an adjustment function between local clocks of different nodes with respect to one reference node. The advantages of this protocol are the elimination of the non deterministic sources of error that affect the protocol itself and the synchronization of the whole network with respect to one reference node while maintaining a minimal control overhead.

*MOTS-CLÉS:* Synchronisation, Réseaux Multi-Sauts, Protocoles distribués, Nœud MPR, Ensemble dominant

*KEYWORDS:* Synchronization, Multi-hop Networks, Distributed Protocols, MPR, Dominating Set

---

## 1. Introduction

La synchronisation des nœuds est une question critique dans les réseaux à architecture distribuée. Pendant plusieurs années, le protocole NTP (Mills, 2006) a réussi à mettre en phase les horloges de l'Internet. Cependant, une nouvelle classe de réseaux a émergé grâce au développement et la diffusion des composants de communications sans fil tel que les routeurs et les capteurs. Des réseaux composés de telles technologies peuvent être déployés d'une manière ad hoc où les communications peuvent être multi-sauts. Les nœuds de ces réseaux ont besoin de coopérer pour assurer les services prévus. La nature partagée et limitée des liens radio qui caractérisent ces réseaux nous amène à concevoir des protocoles légers en terme d'utilisation de la bande passante. Ces réseaux ont besoin d'une synchronisation précise - parfois de l'ordre de la microseconde - pour assurer des applications collaboratives et respecter certaines contraintes d'énergie et de sécurité.

Pendant plusieurs années, un certain nombre d'algorithmes de synchronisation des réseaux a été proposé. Ces algorithmes cherchent à minimiser l'erreur de synchronisation due principalement aux délais indéterministes de transmission de messages entre deux nœuds. Cependant, la plupart de ces algorithmes partage la même architecture de base : un client estime l'écart de son horloge par rapport à celle du serveur en échangeant avec ce dernier un certain nombre de messages. Notre travail explore une forme de synchronisation du temps qui diffère des modèles classiques. Il se repose sur le protocole RBS (Elson *et al.*, 2002) où les nœuds dans la portée de transmission d'un émetteur enregistrent les temps locaux de réception de messages de références envoyés par cet émetteur. Ensuite, ces récepteurs s'échangent leurs temps respectifs de réception pour estimer les écarts d'horloges entre eux.

Notre approche propose une extension du protocole RBS qui permet de synchroniser de manière efficace tous les nœuds d'un réseau sans fil multi-sauts avec un seul nœud de référence. Un ensemble de nœuds participant au mécanisme de synchronisation sera sélectionné dynamiquement parmi l'ensemble de nœuds MPR<sup>1</sup> (Qayyum *et al.*, 2002). Ces nœuds constitueront un ensemble dominant qui garantit que tous les nœuds du réseau soient synchronisés par rapport au nœud de référence.

Dans cet article, nous proposons un mécanisme de synchronisation qui décrit les échanges de messages entre les différents nœuds du réseau. Il sera organisé de la manière suivante. Dans la section 2, nous introduirons les mécanismes pertinents de synchronisation, les sources d'erreurs qui les affectent, le modèle d'horloge considéré, les objectifs de notre approche et ses domaines d'application. Dans la section 3, nous présenterons en détail notre mécanisme de synchronisation, ses composants et nous montrerons son bon fonctionnement. Pour conclure, nous présenterons dans la section 4 nos perspectives et nos travaux en cours.

---

1. La notion des Relais MultiPoint est introduite par le protocole OLSR pour réduire la surcharge des échanges de messages par rapport au mécanisme classique d'inondation. L'ensemble des nœuds MPR d'un nœud donné permet d'atteindre tous les voisins à 2 sauts de ce nœud.

## 2. Les concepts de base

### 2.1. Le modèle d'horloge

Les horloges utilisées dans les composants électroniques modernes sont caractérisées par deux importants paramètres :

- La précision : c'est la différence entre les fréquences d'oscillation théorique et réelle. Cet écart est appelé erreur de fréquence, il est donnée par le fabricant de l'oscillateur et il est généralement de l'ordre de  $10^{-6}$  pour les oscillateurs utilisés dans les composants électroniques modernes.

- La stabilité : c'est la tendance de l'oscillateur à garder la même fréquence avec le temps. L'instabilité de l'oscillateur peut être à court terme et due aux facteurs d'environnement tels que les variations de la température, de la pression, de la tension d'alimentation etc., ou à long terme et due à des effets plus subtils tels que le vieillissement de l'oscillateur.

L'instabilité de la fréquence d'oscillations entraîne une variation continue de la différence de phases entre les horloges des différents nœuds du réseau. En général, les algorithmes de synchronisation des réseaux (eg : NTP) ajustent les erreurs de phase et de fréquence de l'horloge mais n'essaient pas de modéliser la variation continue de la fréquence. Ceci signifie que les ajustements de fréquence sont faits d'une manière continue. Ainsi, nous nous reposons sur une modélisation de l'horloge du système comme ayant une stabilité de fréquence élevée à court terme.

### 2.2. Les sources d'erreurs de synchronisation des horloges

La cause majeure d'erreur dans les protocoles de synchronisation provient de l'estimation indéterministe de la latence des messages échangés. Pour mieux comprendre ces causes d'erreurs, il est utile de décomposer les sources de latence des messages. Les auteurs dans (Kopetz *et al.*, 1989) les divisent en quatre catégories différentes :

- Le temps d'envoi : il s'agit du temps utilisé par l'émetteur pour construire le message. Ce temps inclut les délais de traitement du noyau et ceux introduits par le système d'exploitation.

- Le temps d'accès : il s'agit du temps d'attente pour accéder au canal de transmission. Ce temps est spécifique au protocole MAC utilisé. La couche MAC du 802.11(802.11, 1999) utilise un algorithme distribué qui retarde l'accès au canal d'un temps aléatoire borné. De plus, les retransmissions au niveau MAC dues aux problèmes d'interférence ou de collisions font augmenter ce temps d'accès.

- Le temps de propagation : il s'agit du temps nécessaire pour que le message transite de l'émetteur au(x) récepteur(s). Dans les réseaux sans fil où la portée est limitée à quelques centaines de mètres, le temps de propagation est négligeable.

- Le temps de réception : il s'agit du temps de traitement nécessaire à l'interface réseau du récepteur pour la réception du message et la notification au système de son

arrivée. Ceci est typiquement le temps nécessaire à une interface réseau pour générer un signal de réception d'un message.

### **2.3. Les travaux sur la synchronisation**

Les travaux de Lamport (Lamport, 1978) constituent une référence dans le domaine de la synchronisation qui éclaire l'importance des horloges virtuelles dans les systèmes où la causalité est plus importante que le temps absolu. Lamport ne donne qu'un ordre global aux messages échangés et ne mesure pas les écarts temporels entre eux. Ce travail possède plusieurs applications surtout dans les réseaux de capteurs où l'on a juste besoin d'un temps relatif.

Pendant des années, beaucoup de protocoles ont été conçus pour la synchronisation des horloges dans des réseaux informatiques (Lai *et al.*, 2003), (So *et al.*, 2004). Tous ces protocoles ont le même mécanisme de base : un échange d'informations des horloges entre des clients et un ou plusieurs serveurs, d'où leur appellation émetteur-à-récepteur. Les méthodes pour atténuer les effets de l'indéterminisme dans la livraison de messages et leurs traitements peuvent différer. Le protocole NTP atteint cette atténuation en augmentant le nombre de requêtes qui rendent plus probable l'obtention d'une requête qui subit des délais minimaux. Timing-Sync (Ganeriwal *et al.*, 2003), lui, l'atteint en se basant sur l'estampillage au niveau physique. Ces mécanismes peuvent assurer une *Synchronisation externe* si le réseau est compatible ou synchrone à une norme externe (eg. GPS) ou une *Synchronisation interne* si le serveur est considéré comme l'horloge canonique ou simplement un arbitre des horloges des clients.

En revanche, les protocoles de synchronisation récepteurs-à-récepteurs, comme RBS et (PalChaudhuri *et al.*, 2004), éliminent le temps d'envoi et le temps d'accès, sources majeures des délais, en exploitant la nature broadcast du médium sans fil. Pour un ensemble de récepteurs, le temps d'envoi et le temps d'accès sont identiques pour un paquet envoyé par une source. Les seuls délais variables sont le temps de propagation et le temps de réception. Le premier est infiniment petit et supposé nul par la suite et le deuxième est négligeable par rapport aux temps d'envoi et d'accès. Cette approche entraîne la synchronisation des différents récepteurs l'un par l'autre contrairement à la synchronisation émetteur-à-récepteur.

### **2.4. L'objectif et les domaines d'applications de notre approche**

L'objectif de cet article est de proposer un mécanisme de synchronisation des nœuds d'un réseau par rapport à un nœud de référence. Ce nœud peut être choisi par rapport à des préférences administratives (eg : un nœud avec une synchronisation GPS au UTC ou une passerelle vers Internet dans un réseau Mesh) ou par un algorithme d'élection. La manière de la sélection de ce nœud ne sera pas traitée dans cet article ; nous supposons que ce nœud de référence existe et qu'il est unique. Cette syn-

chronisation est faite grâce à une estimation de l'offset et de la déviation de l'horloge locale d'un nœud par rapport au temps de référence (celui du nœud de référence). Notre mécanisme a comme but de fournir une *fonction d'ajustement* pour transformer les estampilles temporelles générées par l'horloge locale  $T_{local}$  en estampilles synchronisées  $T_{sync}$ . Cette fonction d'ajustement est définie par la formule suivante :

$$T_{sync} = deviation * T_{local} + offset$$

L'objectif est, par conséquent, d'estimer continuellement les paramètres de déviation et d'offset.

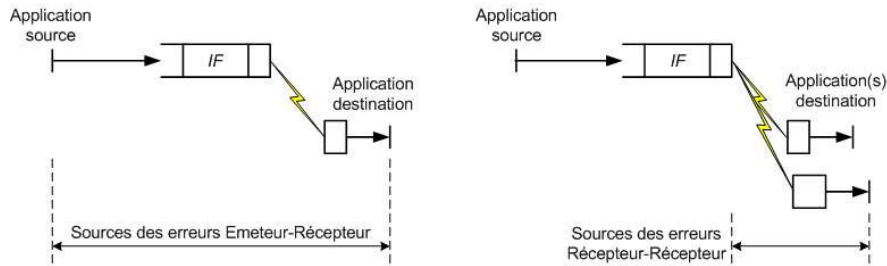
Le mécanisme décrit dans cet article est conçu pour les réseaux sans fil multi-sauts stables tel que les réseaux Mesh. Un mécanisme pour sélectionner les nœuds MPR est supposé mis en place. Des messages *HELLOs*, contenant la liste des voisins directs, sont envoyés périodiquement. Ces messages permettent à chaque nœud de connaître son voisinage à deux sauts et de choisir ses nœuds MPR par la suite. Cela peut être réalisé avec l'utilisation d'un protocole de routage tel que OLSR (Clausen *et al.*, 2003) ou des extensions de DYMO (Chakeres *et al.*, 2006). L'utilisation des nœuds MPR pour transmettre les paquets de diffusion est appropriée pour les réseaux larges et denses. Seuls les nœuds MPR retransmettent un paquet de diffusion d'où une optimisation considérable en terme de paquets de contrôle.

### 3. Le protocole de synchronisation

Dans cette partie, nous introduisons la technique de synchronisation de récepteur-à-récepteur sur laquelle notre protocole se base. Ensuite nous décrivons le mécanisme fondamental de notre protocole de synchronisation et son principe de fonctionnement à un saut puis en multi-sauts.

#### 3.1. La synchronisation de récepteur-à-récepteur

Le mécanisme de synchronisation RBS sur lequel est basé notre approche exploite la nature de diffusion du médium sans fil. Grâce à cette propriété, les nœuds dans la portée de transmission du même émetteur reçoivent le même message avec un très faible décalage. En considérant seulement les temps de réception des différents récepteurs, le protocole RBS élimine immédiatement deux importantes sources d'indéterminisme impliquées dans les transmissions de messages, à savoir le temps d'envoi et le temps d'accès au canal. Les seules sources d'indéterminisme qui interfèrent dans la synchronisation RBS sont le temps de propagation et le temps de réception très inférieur à l'erreur de synchronisation dans le cas d'une synchronisation émetteur-à-récepteur. La Figure 1 illustre cette minimisation de sources d'erreurs dans les protocoles de type récepteurs-à-récepteurs.



**Figure 1.** Sources indéterministes d'erreurs pour la synchronisation récepteur-à-récepteur vs. émetteur-à-récepteur

Le protocole de synchronisation récepteur-à-récepteur RBS fonctionne de la manière suivante : un nœud émetteur transmet un message de référence à n'importe quel instant. Les nœuds dans la portée de transmission de cet émetteur enregistrent le temps local de réception de ce message. Ensuite, les récepteurs s'échangent entre eux les temps de réception enregistrés. Chaque récepteur peut estimer l'offset de son horloge par rapport à celle d'un autre récepteur. Les auteurs de RBS ont pu estimer la distribution de l'erreur de synchronisation pour l'ensemble des récepteurs. Ainsi, une suite de messages de référence est envoyée de l'émetteur à l'ensemble de récepteurs. La différence entre les temps réels de réception des messages donne une distribution gaussienne qui a une moyenne égale à zéro.

Le mécanisme de base de RBS est conçu pour les réseaux où l'émetteur des messages de références est voisin de tous les autres nœuds du réseau. Étendre ce mécanisme de synchronisation pour les réseaux sans fil multi-sauts (eg. Réseau Mesh) paraît important et utile à la fois. Or la transmission dans ce genre de réseau peut générer un nouveau degré de difficulté pour l'estimation de la variation des temps de réception du message. Pour éviter la perte de précision, dans RBS, deux nœuds situés dans différents voisinages sont typiquement synchronisés en utilisant un troisième nœud se trouvant dans l'intersection des deux voisinages.

Malgré les avantages de l'élimination des sources majeures de l'indéterminisme des erreurs et le degré de précision de synchronisation qui en découle, le mécanisme RBS comporte certaines limitations :

- 1) Il nécessite que les récepteurs des messages de référence émis par l'émetteur échangent entre eux les temps de réceptions. Pourtant, il est possible que deux nœuds récepteurs ne soient pas voisins et n'aient pas, par conséquent, la possibilité d'être synchronisés entre eux. RBS ne prend pas ce cas en considération.
- 2) Dans la synchronisation multi-sauts plusieurs émetteurs sont nécessaires pour assurer la synchronisation. RBS ne donne aucun mécanisme de choix de ces nœuds.
- 3) Dans RBS, chaque récepteur doit fournir le temps de réception à tous les émetteurs de son voisinage. Cet exigence qui augmente en  $O(N^2)$  le nombre de messages

généérés, peut être évitée surtout dans le cas de la synchronisation par rapport à une référence comme nous allons voir dans la suite.

### 3.2. *Le mécanisme de synchronisation à un saut*

Dans un réseau à un seul saut où tous les nœuds sont voisins les uns des autres, notre mécanisme de synchronisation comporte deux étapes principales : (1) la sélection des nœuds émetteurs et (2) la synchronisation des nœuds.

#### 3.2.1. *La sélection des nœuds émetteurs*

Dans un réseau à un saut, la synchronisation des différents nœuds par rapport au nœud de référence nécessite deux émetteurs de messages de références. Le premier est le nœud de référence lui-même  $N_R$  et le deuxième est choisi de la manière suivante : le nœud de référence  $N_R$  transmet un message d'avertissement  $M_{avr}$  pour indiquer sa présence et pour démarrer le processus de synchronisation. Le message  $M_{avr}$  contient l'adresse IP du  $N_R$  et un numéro de séquence, ces informations sont uniques pour un message donné et permettent d'identifier la fraîcheur du message reçu. A la réception de ce message, chaque nœud choisit un temps aléatoire borné avant d'essayer de retransmettre le  $M_{avr}$ . Le premier nœud qui réussit à transmettre ce message (celui qui a choisi le temps aléatoire le plus court) est considéré comme le deuxième nœud émetteur  $N_E$ . A la réception du deuxième message d'avertissement, les autres nœuds annulent leur émission du message  $M_{avr}$  et se considèrent clients du mécanisme de synchronisation.

#### 3.2.2. *La synchronisation des nœuds*

Le nœud émetteur  $N_E$  initie le cycle de synchronisation en répétant les étapes suivantes :

1) Pendant un cycle de synchronisation, le nœud émetteur transmet  $n$  messages de référence  $M_{ref}$ . Chaque message contient le numéro du message dans le cycle, le nombre total de messages à émettre pendant le cycle et un numéro de séquence qui est incrémenté à chaque envoi du message de synchronisation pour le différencier des autres. Ces informations permettent à chaque nœud de détecter la fin de la transmission des messages de référence.

2) Chaque récepteur, y compris le nœud de référence, enregistre les temps de réception et le numéro de séquence des messages reçus. Une fois le cycle de transmission terminé, le nœud construit un message de réponse  $M_{rep}$  qui contient les temps de réception et l'envoie au nœud émetteur.

Le nœud émetteur reçoit tous les temps de réception envoyés par les différents récepteurs. Puis il calcule les fonctions d'ajustement  $F_{aju}$  obtenues en utilisant la méthode de la régression linéaire qui lie les estampilles temporelles des différents nœuds à celles du nœud de référence. Cette méthode permet non seulement d'estimer l'offset entre les horloges des nœuds et celle du nœud de référence mais en plus elle

permet d'estimer la variation de la fréquence de l'horloge d'un nœud par rapport à celle du nœud de référence.

Le nœud émetteur diffuse ensuite un message d'ajustement  $M_{aju}$  qui contient les adresses de chaque récepteur ainsi que sa fonction d'ajustement  $F_{aju}$ . Cette fonction d'ajustement sera utilisée par la suite pour transformer les estampilles générées avec l'horloge locale d'un nœud en estampilles synchronisées.

Une fois que le  $M_{aju}$  est reçu par le nœud de référence, il commence son cycle de transmission de  $M_{ref}$  afin de synchroniser le nœud émetteur  $N_E$ , seul nœud non synchronisé du réseau. Ce cycle est identique à celui décrit au dessus avec l'unique différence que seuls les nœuds  $N_E$  et celui avec la plus petite adresse IP répondront aux messages de référence. La raison de cette procédure est de minimiser la génération des messages de synchronisation. A la fin de ce cycle, tous les nœuds du réseau seront synchronisés.

### 3.2.3. Une légère charge de synchronisation

Ce mécanisme de synchronisation à un saut permet de réduire le nombre de messages générés par rapport au mécanisme de base décrit dans RBS. Selon notre mécanisme, le cycle de synchronisation nécessite en total  $O(N)$  messages par rapport à  $O(N^2)$  messages pour RBS. Cette réduction est due au fait que dans notre mécanisme seul le nœud émetteur est responsable de calculer les fonctions d'ajustement  $F_{aju}$ . Les récepteurs n'ont pas besoin d'échanger entre eux les messages  $M_{rep}$ .

Notre mécanisme permet aux nœuds récepteurs d'attendre la réception de plusieurs messages  $M_{ref}$  avant l'envoi du message de réponse  $M_{rep}$  contenant les différents temps de réception. Cette variation par rapport au mécanisme de base RBS permet de minimiser l'utilisation de la bande passante du réseau.

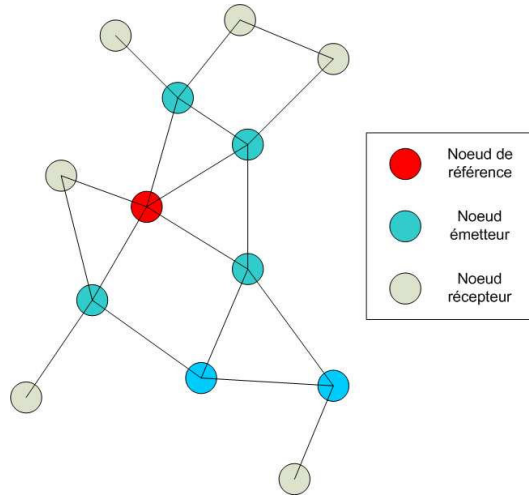
## 3.3. Le mécanisme de synchronisation multi-sauts

Dans un réseau multi-sauts notre mécanisme de synchronisation comporte aussi deux étapes principales. La première consiste à sélectionner un ensemble de nœuds émetteurs qui doit couvrir tous les nœuds du réseau c'est-à-dire que chacun de ces nœuds doit avoir au moins un voisin appartenant à cet ensemble qu'on appelle ensemble dominant  $DS$  (du point de vue de la théorie des graphes). La deuxième étape consiste à synchroniser les nœuds par rapport au nœud de référence.

### 3.3.1. La sélection des nœuds émetteurs

Le nœud de référence  $N_R$  diffuse un message d'avertissement  $M_{avr}$  à travers tout le réseau pour indiquer sa présence et démarrer le processus de synchronisation. Le message  $M_{avr}$  contient l'adresse IP du nœud de référence, un numéro de séquence et le nombre de sauts traversés. Chaque nœud décide, se basant sur ce message et sur





**Figure 2.** Rôles des nœuds dans le mécanisme de synchronisation dans un réseau multi-sauts

ses informations locales (ses nœuds MPR et ses MPR selectors<sup>2</sup>) s'il compte parmi l'ensemble des nœuds émetteurs ou des nœuds récepteurs.

Cette décision respecte les règles suivantes :

- 1) Les messages  $M_{avr}$  avec un numéro de séquence plus petit par rapport à celui déjà traité sont rejetés. Par la suite, nous considérons comme message récent le message dont le numéro de séquence est plus grand ou égal à celui déjà traité.
- 2) Les messages  $M_{avr}$  récents avec un nombre de sauts plus grand ou égal à celui déjà connu sont rejetés. Ceux qui ont un nombre de sauts plus petits sont traités.
- 3) Si le message  $M_{avr}$  n'arrive pas d'un MPR selector, le nœud se considère comme récepteur dans le mécanisme de synchronisation. Ce nœud ne répondra qu'à la source de ce message qui sera considéré comme son *émetteur maître*.
- 4) Si le message  $M_{avr}$  arrive d'un MPR selector, le nœud se considère comme émetteur dans le mécanisme de synchronisation. Un nœud émetteur  $N_E$  répond aux messages de référence  $M_{ref}$  envoyés par le MPR selector, source de ce message, et à ses nœuds MPR. Cette exigence permet d'étendre le mécanisme de synchronisation par récurrence saut par saut tout en limitant le nombre des messages échangés.  $N_E$  devra relayer les  $M_{avr}$  en incrémentant le champ du nombre de saut.

La Figure 2 illustre un simple réseau multi-sauts et donne le rôle de chaque nœud suivant le mécanisme de choix des nœuds émetteurs décrit ci-dessus.

<sup>2</sup> L'ensemble des MPR selectors d'un nœud X est tout nœud Y voisin direct de X qui a choisi X comme nœud MPR.

**Propriété :** l'ensemble des nœuds émetteurs  $\{N_{Ei}\}$  sélectionnés par ce mécanisme forme un ensemble connecté dominant *CDS* (Adjih *et al.*, 2002).

*Preuve :*

*Propriété de connectivité :* L'ensemble des nœuds sélectionnés est formé par :

Instance 1 : les nœuds MPR du nœud de référence,

Instance 2 : des nœuds MPR des nœuds choisis dans l'instance 1 et qui sont à deux sauts du nœud de référence,

...

Instance  $i$  : des nœuds MPR des nœuds choisis dans l'instance  $i-1$  et qui sont à  $i$  sauts du nœud de référence.  $i \leq r$  où  $r$  est le rayon du réseau.

Cet ensemble est par construction connecté.

*Propriété de couverture :* Essayons de prouver par l'absurde que l'ensemble des nœuds émetteurs  $\{N_E\}$  n'est pas dominant. C'est à dire qu'il existe des nœuds qui n'appartiennent pas à  $V(N_E)^3$ .

Considérons l'ensemble  $G$  des nœuds permettant de connecter quelques nœuds de  $\{N_E\}$  à quelques nœuds de  $\overline{V(N_E)}$  complément de  $V(N_E)$ .  $G$  est l'ensemble des nœuds qui ont au moins un voisin dans  $\{N_E\}$  et au moins un voisin dans  $\overline{V(N_E)}$ . Comme le réseau est supposé connecté notre hypothèse engendre que  $G$  n'est pas vide. Remarquons que  $G \cap \{N_E\} = \emptyset$  par construction.  $G$  a au moins un voisin dans  $\overline{V(N_E)}$  qui est par conséquent un voisin à 2 sauts d'un nœud  $N_E$ . Or l'ensemble des MPR d'un nœud lui permet d'atteindre tous ses voisins à deux sauts. Ainsi il existe un nœud appartenant à  $G$  qui a un  $N_E$  comme MPR selector. Ce qui signifie que ce nœud sera choisi comme émetteur pour le mécanisme de synchronisation et ce nœud appartient à  $G \cap \{N_E\}$ , vide par construction. Ce qui contredit notre supposition.

### 3.3.2. La synchronisation des nœuds

Pour les réseaux multi-sauts, la synchronisation est progressive, c'est-à-dire que les nœuds au  $i^{eme}$  saut ne se synchronisent qu'après ceux au  $(i - 1)^{eme}$  saut. Elle se base sur le même mécanisme décrit dans la section 3.2.2 à quelques différences près. En effet, un récepteur ne traite, dans cette approche, que les messages de référence envoyés par son émetteur maître. Ce qui permet de minimiser la charge de communication dans le réseau tout en conservant la synchronisation de tous les nœuds. De plus, un récepteur ne tient compte que des fonctions d'ajustement proposées par un seul émetteur ce qui induit à un minimum d'incohérence.

On définit  $N_{Ei}$  comme un nœud émetteur à  $i$  saut(s) par rapport au nœud de référence noté  $N_{E0}$ . La synchronisation des nœuds au premier saut (par rapport au nœud de référence) se fait de la manière suivante :

1) Les nœuds  $N_{E1}$  (nœuds émetteurs du premier saut) lancent le cycle de synchronisation décrit dans la section 3.2.2. A la fin de cette étape tous les nœuds à 1 et 2 sauts

3. ensemble des nœuds qui ont au moins un voisin émetteur.

du nœuds de référence seront synchronisés à l'exception des nœuds  $N_{E1}$  qui n'ont pas des voisins dans l'ensemble des nœuds  $\{N_{E1}\}$ .

2) Le nœud de référence  $N_{E0}$  lance le cycle de synchronisation qui résulte de la synchronisation de tous les nœuds non synchronisés dans la première étape.

3) A partir du deuxième saut, chaque nœud émetteur  $N_{Ei}$  pourra commencer son cycle de synchronisation après la réception d'un message d'ajustement de l'un des nœuds  $N_{Ei-1}$ .

A la fin de ce mécanisme de synchronisation chaque nœud dans le réseau sera synchronisé par rapport au nœud de référence. Cette synchronisation globale du réseau est assuré par le fait que l'ensemble des nœuds émetteurs forme un ensemble dominant et qu'à la  $i^{eme}$  itération permettant de synchroniser les nœuds à  $i$  sauts, les nœuds à  $i - 1$  sauts sont déjà synchronisés.

#### 4. Conclusion et perspectives

Le mécanisme de synchronisation décrit dans cet article est basé sur le concept de synchronisation récepteur-à-récepteur. Selon ce mécanisme, un nœud émetteur diffuse périodiquement un message de référence. Ces voisins vont l'informer du temps de réception de ce message. Ainsi, l'émetteur peut mesurer le décalage d'horloge entre ses différents voisins et calculer la fonction d'ajustement de temps pour les synchroniser. La sélection des nœuds émetteurs se fait suivant une démarche précise qui permet de minimiser les échanges de messages et aboutir à la synchronisation de tous les nœuds du réseau par rapport à un seul nœud de référence défini.

Cette synchronisation pourrait être exploité par des applications collaboratives et par des mécanismes de sécurité. L'une des applications directe serait de permettre de corréler des traces d'exécution de certains protocoles de routage ad hoc collectés sur différents nœuds du réseau pour tester le bon fonctionnement du protocole et le valider ou de vérifier des propriétés de sécurité en détectant d'éventuelles anomalies ou attaques.

Ce protocole de synchronisation est en cours d'évaluation et d'implantation. Plusieurs modules du code ont déjà subi un test unitaire. Nous envisageons d'appliquer notre mécanisme sur une plateforme réelle pour évaluer ses performances en terme de précision de synchronisation et de charge du réseau. Nous envisageons également d'étudier l'effet de certains paramètres qui puissent influencer notre mécanisme de synchronisation principalement la mobilité et la défaillance des nœuds.

## 5. Bibliographie

- 802.11 S., « Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications », ANSI/IEEE, 1999.
- Adjih C., Jacquet P., Viennot L., Computing Connected Dominating Sets with Multipoint Relays, Technical Report n° 4597, INRIA Rocquencourt, October, 2002.
- Chakeres I., Perkins C., « Dynamic MANET On-demand (DYMO) Routing », Intrnet Draft, draft-ietf-manet-dymo-06.txt, October, 2006. Work in progress.
- Clausen T., Jacquet P., « Optimized Link State Routing Protocol (OLSR) », RFC 3626, October, 2003.
- Elson J., Girod L., Estrin D., « Fine-Grained Network Time Synchronization Using Reference Broadcasts. », *OSDI*, 2002.
- Ganeriwal S., Kumar R., Srivastava M. B., « Timing-sync protocol for sensor networks. », *SenSys*, p. 138-149, 2003.
- Kopetz H., Schwabl W., Global time in distributed real-time systems, Technical Report n° 15/89, Technische Universitat Wien, 1989.
- Lai T.-H., Zhou D., « Efficient and Scalable IEEE 802.11 Ad-Hoc-Mode Timing Synchronization Function », *AINA*, IEEE Computer Society, p. 318-323, 2003.
- Lamport L., « Time, Clocks, and the Ordering of Events in a Distributed System », *Commun. ACM*, vol. 21, n° 7, p. 558-565, 1978.
- Mills D. L., *Computer Network Time Synchronization : the Network Time Protocol*, CRC Press, 2006.
- PalChaudhuri S., Saha A. K., Johnson D. B., « Adaptive clock synchronization in sensor networks », in , K. Ramchandran, , J. Sztipanovits, , J. C. Hou, , T. N. Pappas (eds), *IPSN*, ACM, p. 340-348, 2004.
- Qayyum A., Viennot L., Laouti A., « Multipoint Relaying for Flooding Broadcast Messages in Mobile Wireless Networks. », *HICSS*, p. 298, 2002.
- So J., Vaidya N. H., A Distributed Self-Stabilizing Time Synchronization Protocol for Multi-Hop Wireless Networks, Technical report, University of Illinois at Urbana-Champaign, January, 2004.