Gestion de la mobilité dans les réseaux de capteurs sans fil

Damien Roth

Numéro d'ordre : 1742

UNIVERSITÉ DE STRASBOURG

École Doctorale Mathématiques, Sciences de l'Information et de l'Ingenieur

THÈSE

présentée pour obtenir le grade de

Docteur de l'Université de Strasbourg
Discipline : Informatique

par

Damien ROTH

Gestion de la mobilité dans les réseaux de capteurs sans fil

Soutenue publiquement le 23 novembre 2012

Membres du jury :

Directeur :
THOMAS NOËL, Professeur à l'Université de Strasbourg

Co-encadrant :
JULIEN MONTAVONT, Maître de Conférences à l'Université de Strasbourg

Rapporteurs :
ISABELLE CHRISMENT, Professeur à l'Université de Lorraine
CONGDUC PHAM, Professeur à l'Université de Pau

Examinateur :
LAURENT TOUTAIN, Maître de Conférences à Telecom Bretagne

Laboratoire des Sciences de l'Image, de l'Informatique et de la Télédétection CNRS UMR 7005

La connaissance scientifique possède en quelque sorte des propriétés fractales : nous aurons beau accroître notre savoir, le reste - si infime soit-il - sera toujours aussi infiniment complexe que l'ensemble de départ.

— Isaac Asimov

REMERCIEMENTS

Le travail réalisé tout au long de cette thèse n'a été possible qu'avec l'aide et le soutien de nombreuses personnes. Je profite de l'occasion qui m'est donnée ici pour leur exprimer toute ma gratitude.

Je tiens tout d'abord à remercier Thomas Noël et Julien Montavont pour leur encadrement, leur disponibilité et les nombreux et précieux conseils qu'ils m'ont prodigué au cours de ces trois années de thèse.

Je remercie également Isabelle Chrisment, professeur à l'Université de Lorraine, et Congduc Pham, professeur à l'Université de Pau, d'avoir montré leur intérêt pour mes travaux en acceptant la charge de rapporteur. Je remercie également Laurent Toutain, maître de conférence à Télécom Bretagne, d'avoir bien voulu juger ces travaux en tant qu'examinateur.

Je voudrais également remercier tous les membres de l'équipe Réseaux et Protocoles du LSIIT. En particulier les doctorants présents et passés, avec qui j'ai beaucoup appris, partagé et apprécié travailler. Merci à Vincent, Romain, Julien, Randa, François et Oana pour votre disponibilité et votre gentillesse. Finalement, un grand merci à Erkan pour ses nombreux conseils techniques sur la programmation embarquée.

Je remercie ma famille et mes amis pour leur irremplaçable et inconditionnel soutien. Ils ont également une part importante dans la réalisation de cette thèse.

Enfin, bien sûr, je remercie Emeline qui m'a soutenu et supporté durant toute cette thèse et surtout durant la dernière et longue, interminable ligne droite.

TABLE DES MATIÈRES

INTRODUCTION GÉNÉRALE

INTRODUCTION GÉNÉRALE

LES RÉSEAUX DE CAPTEURS SANS FIL

La miniaturisation croissante des équipements électroniques ainsi que les progrès des technologies de communication sans fil ont permis de produire à faible coût et en grande quantité des nœuds capteurs communicants et peu consommateurs en énergie (figure 1). Ces petites entités autonomes mesurent les conditions ambiantes (luminosité, température, pression barométrique, son, ondes sismiques, etc.) et les transforment en signaux électriques permettant aux équipements informatiques de les traiter [1]. Bien qu'il une large variété de nœuds capteurs sans fil, leur architecture matérielle reste similaire. Un capteur est composé :

- d'une ou plusieurs unités de capture, chargées de récolter les informations à propos de l'environnement proche ;
- d'une unité de traitement : un processeur et de la mémoire ;
- d'une batterie ;
- d'un module de transmission sans fil.

De par leur petite taille, ces équipements sont extrêmement contraints. Individuellement, ils ne peuvent pas rivaliser avec les ordinateurs ou terminaux de poche actuels en terme de traitement des données, de capacité de stockage et de communication. La table 1 présente les principales caractéristiques de quelques nœuds capteurs sans fil. Comme nous pouvons le constater, la cadence des processeurs des nœuds capteurs est bien inférieure à celle des derniers terminaux de poche apparus sur le marché (8 ou 16 MHz contre plusieurs cœurs et une vitesse dépassant 1 Ghz).

La force des réseaux de capteurs ne repose pas sur les capacités d'un seul équipement mais provient de leur capacité à collaborer. Le faible coût des nœuds capteurs permet d'en déployer un grand nombre pour couvrir de larges zones. Toutefois, un tel déploiement peut être problématique pour récolter les données, en particulier si les nœuds capteurs sont déployés dans un environnement inhospitalier. Pour faciliter cette récolte, les

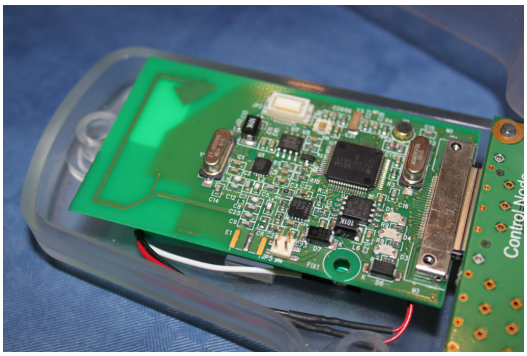


FIGURE 1: Nœud capteur sans fil WSN 430 de la plateforme SensLAB

nœuds capteurs peuvent former un réseau grâce à leur interface de communication. Les données obtenues par les capteurs seront généralement transmises à une entité spécialisée nommée puits.

Un puits peut avoir plusieurs fonctions au sein d'un réseau de capteurs. Il peut être un équipement collecteur et disposer d'une puissance de calcul plus élevée ainsi que d'une mémoire de grande capacité pour traiter directement les données reçues. Dans ce cas, un utilisateur pourra donc récupérer manuellement les données à un endroit unique. Le puits peut également disposer d'une interface de communication supplémentaire permettant de le relier aux réseaux standards (réseau privé ou Internet). Il offre ainsi la possibilité aux utilisateurs d'accéder à distance aux données qu'il contient ou permettre de trans-

Nœud capteur	MicaZ	TelosB	WSN430
Processeur	Atmel AT -Mega 128L	TI MSP430	TI MSP430
Vitesse du processeur	16 MHz	8 MHz	8 MHz
RAM	4 Ko	10 Ko	10 Ko
Espace programme	128 Ko	48 Ko	48 Ko
Radio	TI CC2420 IEEE 802.15.4		TI CC1100
Fréquence (MHz)	2400-2483		315/433/868/915
Débit (Kb/s)	250		76.8
Batterie	2 x AA	2 x 3A	PoLiFlex
Voltage	2,7 V	1,8 - 3,6 V	3,7 V
Dimension (mm)	58x32x7	36x48x9	65x40x8

TABLE 1: Caractéristiques de quelques nœuds capteurs sans fil

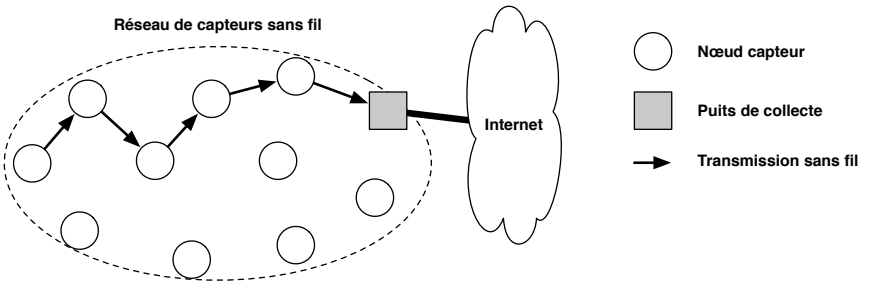


FIGURE 2: Représentation schématique d'un réseau de capteurs sans fil collaborant pour acheminer des données vers le puits

mettre directement les données des nœuds capteurs à une base de stockage distante.

La portée de transmission des nœuds capteurs peut aller de quelques mètres à plusieurs centaines de mètres. Cette portée va dépendre de cinq conditions : la localisation (présence d'obstacles, par exemple), la fréquence, la modulation, la puissance d'émission et les conditions météorologiques [2]. De ce fait, en fonction de leur déploiement, tous les nœuds ne seront pas forcément à portée radio du puits. Ils vont ainsi devoir collaborer pour envoyer de proche en proche les informations récoltées jusqu'au puits. Cette communication multi-sauts est illustrée dans la figure 2. Les conditions de déploiement vont alors poser de nouvelles contraintes et défis.

Les communications multi-sauts vont nécessiter la mise en place d'un protocole de routage des données afin de gérer les multiples transmissions pour atteindre le destinataire du paquet de données. Cette gestion n'est pas aisée car la topologie d'un réseau de capteurs sans fil est dynamique : un nœud capteur peut ne plus être joignable parce qu'il a épuisé toutes ses réserves énergétiques ou parce que les conditions environnementales ont fortement détérioré ses capacités de communication. Ces protocoles de routage doivent également être économes en énergie car les réseaux de capteurs sont conçus pour être déployés sur des durées allant de quelques mois à plusieurs années. De plus, les conditions de déploiement peuvent rendre l'accès aux capteurs difficiles. Il sera donc impossible de remplacer ou reconfigurer les nœuds une fois que ces derniers seront déployés.

Trafic de données et applications

Dans les réseaux de capteurs sans fil, le trafic de données va essentiellement dépendre de la raison pour laquelle le réseau a été déployé. Il est possible de distinguer trois types de trafic de données qui peuvent être utilisés seuls ou conjointement :

Orienté temps (*time-driven*) Le premier type de trafic est un trafic périodique dans lequel les nœuds vont transmettre à intervalle régulier les données obtenues des capteurs (avec une agrégation des valeurs ou non). La fréquence de l'envoi des données va dépendre de l'application installée sur les nœuds et peut aller de quelques millisecondes à plusieurs heures voire plusieurs jours. C'est le trafic de données le plus utilisé dans les réseaux de capteurs sans fil.

Orienté requêtes (*query-driven*) Le second type de trafic est centré sur l'utilisateur. Un utilisateur ou une application (distante ou installée sur le puits) va envoyer une requête à un nœud ou à un groupe de nœuds pour obtenir les informations de leurs capteurs.

Orienté événements (*event-driven*) Le dernier type de trafic est un trafic événementiel. De manière générale, les nœuds vont récupérer périodiquement les données de leurs capteurs et les analyser. Si le nœud détecte un événement ou phénomène physique à partir de ces analyses, il va transmettre l'information au puits. Ensuite, selon l'application installée sur le nœud et sur le puits, cette remontée d'information pourra être périodique ou à la demande du puits pour confirmer l'événement.

La grande diversité des nœuds capteurs a permis d'envisager de nombreuses applications que nous pouvons regrouper en quatre principales catégories :

Surveillance Les réseaux de capteurs peuvent être utilisés pour étudier l'environnement [3, 4, 5] ou la faune. Lorsque ces nœuds capteurs sont utilisés pour enregistrer les données biologiques d'un être vivant [6], ils sont nommés biologgers. Ces déploiements utilisent un trafic de données de type périodique (dans [3], les données sont transmises toutes les 5 minutes et dans [6] toutes les deux heures) ou de type événementiel (par exemple, seulement si les ondes

sismiques dépassent un seuil prédéfini). Ainsi les données collectées par le projet SensorScope [7] ont permis d'améliorer les prédictions d'inondations en détectant les microclimats dans les glaciers.

Pistage Équipés de caméras ou de systèmes de localisation, les réseaux de capteurs peuvent suivre une cible (par exemple, le déplacement d'animaux, d'êtres humains [8, 9, 10], de colis, etc.). Dans ce type d'application, les informations récoltées sont transmises lorsque la cible a été repérée ou si elle se déplace.

Bâtiments intelligents Les réseaux déployés pour ce type d'application peuvent avoir plusieurs objectifs. Ils peuvent être dédiés à l'étude de structures [11, 12] pour détecter des fissures (qui peuvent se produire lors de séismes) ou détecter l'impact de chantiers sur d'anciennes constructions. Il peuvent également être dédiés à l'étude des conditions ambiantes d'une maison [13], d'une industrie [14] ou de navires [15]. Ces déploiements peuvent également permettre de faire des relevés de compteur à distance comme le fait la société Coronis System [16] aux Sable-d'Olonne. Dans le cadre de la domotique, ces réseaux peuvent contrôler les divers équipements de la maison. L'intérêt des réseaux de capteurs sans fil dans ces types de déploiement est l'absence de câbles reliant l'équipement au puits réduisant ainsi fortement le coût de leur installation [15].

Médical Les réseaux de capteurs sont aussi fréquemment employés par les centres médicaux, notamment pour permettre le suivi de patients. Ainsi, les relevés effectués par le personnel médical (pression sanguine, oxygénation, glycémie, etc.) sont complétés par des relevés en temps réel obtenus par les nœuds capteurs sans fil [17, 18]. En outre, un déploiement de réseaux de capteurs au domicile du patient peut permettre son suivi médical à distance, à moindre coût [19].

CONTEXTE DE RECHERCHE

De plus en plus d'applications vont nécessiter la présence de capteurs sur des éléments mobiles. L'intérêt de la mobilité

dans les réseaux de capteurs sans fil est multiple dans la mesure où les capteurs mobiles peuvent permettre d'étendre la couverture d'un réseau ou encore obtenir des résultats plus précis [20]. En prenant l'exemple d'une application de surveillance, des capteurs peuvent être placés sur des animaux qui évolueront librement dans un parc où seront également déployés des nœuds fixes. Ces derniers récupèreront les données émises par les nœuds capteurs mobiles. En outre, il est fort probable qu'à l'instar des équipements des réseaux IP, les nœuds mobiles ne soient plus confinés au sein d'un seul réseau mais évolueront au sein de multiples réseaux. Cette mobilité est d'ailleurs un des éléments clés de l'Internet des Objets [21].

Dans les réseaux sans fil, la mobilité d'un équipement fait intervenir plusieurs niveaux de la pile de communication (modèle OSI [22]). Elle se gère tout d'abord au niveau 2, où un protocole d'accès au médium (MAC) doit permettre aux nœuds mobiles de transmettre leurs paquets sur le médium sans fil. Plusieurs verrous scientifiques peuvent limiter les capacités de communication des nœuds mobile au niveau MAC. En effet, un nœud mobile doit être capable de s'insérer dans les communications mais également de déterminer vers qui transmettre ses paquets. De nombreuses solutions facilitant l'insertion de nœuds mobiles ont été proposées dans la littérature [23, 24, 25] mais elles utilisent des informations provenant de protocoles de la couche réseau pour déterminer vers quel voisin transmettre les paquets. En raison du grand nombre de protocoles de routage existants [26], il est fort peu probable que celui utilisé par le nœud mobile soit identique à celui du réseau dans lequel il se trouve. En conséquence, le nœud mobile ne peut obtenir un prochain saut par des moyens conventionnels.

L'arrivée du protocole Internet Protocol version 6 (IPv6) offre de nouvelles perspectives aux réseaux de capteurs sans fil [27, 28]. En disposant d'une adresse IPv6 globale, un nœud mobile peut communiquer directement avec des équipements (correspondants) situés hors du réseau dans lequel le nœud se trouve. Lors de ses déplacements, un nœud mobile peut être amené à changer d'adresse IPv6 et, sans support spécifique, les connexions entre un nœud mobile et ses correspondants devront être réinitialisées. Différentes solutions [29, 30, 31] existent dans les réseaux IP pour pallier ce problème. Cependant, les évaluations faites de ces protocoles au sein des réseaux de capteurs sans fil sont approximatives. Nous pensons qu'il sera né-

cessaire d'étudier et d'évaluer soigneusement ces protocoles avant de proposer une nouvelle solution pour gérer la mobilité au niveau 3.

Le travail présenté dans cette thèse doit répondre à ces questions et proposer également des solutions pour que cette mobilité devienne une réalité pour les réseaux de capteurs sans fil.

GESTION DE LA MOBILITÉ

Lors de l'étude des protocoles MAC, nous nous sommes intéressés aux caractéristiques du médium radio sans fil et, en particulier, à la capacité des nœuds à pouvoir entendre des données, même si elles ne leur sont pas adressées (sur-écoute). En s'appuyant sur cette caractéristique, nous avons proposé une solution au problème de la sélection du prochain saut lors des déplacements de nœuds mobiles. Notre première contribution, le protocole Mobinet, utilise la sur-écoute liée au médium radio sans fil pour découvrir le voisinage d'un nœud mobile et ainsi déterminer vers qui un paquet doit être envoyé. Notre investigation s'est poursuivie avec l'étude des protocoles de support de la mobilité de niveau 3 et nous avons décidé d'évaluer Mobile IPv6 au sein des réseaux de capteurs sans fil.

Durant cette étude, nous démontrons que le mécanisme de détection de mouvement de Mobile IPv6 est inapplicable au sein des réseaux de capteurs sans fil et nous proposerons deux solutions pour pallier ce problème. La première solution est basée sur des temporisateurs afin de rester conforme avec la norme (c'est-à-dire pas de nouveaux mécanismes) et la seconde solution utilise quant à elle le protocole Mobinet. Ces deux solutions ont été implémentées et évaluées sur une plate-forme constituée de nœuds capteurs TelosB déployés dans nos locaux.

CONTRÔLE DE CONGESTION

Au cours de nos travaux, nous avons pu observer que la perte de nombreux paquets de données est liée à l'inaccessibilité du prochain saut. Un prochain saut peut devenir inaccessible pour diverses raisons telles que des perturbations radio, la conges-

tions du médium radio ou une panne. Ce sont les protocoles de routage qui sont chargés de sélectionner un prochain saut. Pour consommer peu d'énergie, ces derniers actualisent leurs informations de routage peu fréquemment. En conséquence, de nombreux paquets peuvent être perdus avant qu'un nouveau prochain saut ne soit désigné par le protocole de routage. Nous nous sommes demandés si la sur-écoute des protocoles MAC peut aider à pallier ce problème. Notre dernière contribution, nommée CLOMAC, est un mécanisme multi-couches qui permet aux nœuds avoisinant un émetteur de devenir la destination de son paquet de données. En créant dynamiquement des chemins alternatifs, notre solution est capable de réduire voire d'éviter les congestions.

PLAN DE LA THÈSE

Ce document se décompose en trois parties. La première partie est consacrée à la présentation du contexte de travail. Nous présenterons les protocoles MAC conçus spécifiquement pour les réseaux de capteurs sans fil et comment ces derniers gèrent la mobilité des nœuds. Nous détaillerons ensuite l'intégration du protocole IPv6 dans les réseaux de capteurs sans fil et les modifications qui y ont été apportées pour prendre en compte les fortes contraintes de ces réseaux (énergétiques et taille de paquets). Enfin, nous ferons un état des différents protocoles de support de la mobilité au niveau 3 existants et leur utilisation au sein des réseaux de capteurs sans fil.

La seconde partie est consacrée à nos travaux sur la gestion de la mobilité qui constituent le cœur de cette thèse. Dans un premier temps, elle présentera notre première proposition permettant à un nœud mobile de sélectionner un prochain saut au niveau MAC lors de ses déplacements. Enfin, nous exposerons les résultats de notre expérimentation de Mobile IPv6 et les moyens mis en œuvre pour pallier le problème de détection de mouvement.

Dans la troisième et dernière partie, nous présenterons nos travaux sur l'évitement de congestion dans les réseaux de capteurs sans fil. Elle fera dans un premier temps l'inventaire des solutions existantes, puis nous détaillerons notre contribution et son évaluation par simulation.

Dans la conclusion, nous exposerons les principales contributions de cette thèse et les mettrons en perspective.

Première partie

CONTEXTE DE RECHERCHE

1

CONTRÔLE D'ACCÈS AU MEDIUM DANS LES RÉSEAUX DE CAPTEURS SANS FIL

Sommaire

1.1 INTRODUCTION

Ce chapitre introduit les fonctions assumées par la couche de contrôle d'accès au médium (Medium Access Control (MAC)). En raison des fortes contraintes des nœuds capteurs, l'utilisation des protocoles MAC des réseaux sans fil existants va être la source de nombreux problèmes liés à la forte consommation de l'émetteur radio. Nous présenterons dans un premier temps ces problèmes de consommation énergétique au sein des réseaux de capteurs sans fil et nous continuerons par la description des principaux concepts (une étude [32] datant de 2009 a recensé plus de 70 protocoles MAC) permettant de les résoudre. Nous conclurons ce chapitre par la présentation de la norme 802.15.4 [23] définie par l'organisme de standardisation Institute of Electrical and Electronics Engineers (IEEE) [33].

1.2 DÉFINITION

La couche MAC est une sous-couche de la couche liaison de données (couche 2) du modèle OSI (Open System Interconnection) [22]. Les protocoles appartenant à cette couche sont chargés de coordonner l'accès au médium de communication qui peut être partagé par de nombreux nœuds. Un protocole MAC conçu pour les réseaux de capteurs sans fil doit donc décider à quel moment un nœud doit entrer dans l'une des trois phases suivantes : mise en veille, transmission et écoute / réception. Ces différentes phases doivent s'alterner tout en essayant de fournir un accès fiable, une faible latence et un débit équitable pour tous les nœuds. Atteindre ces conditions est d'autant plus difficile compte tenu des caractéristiques du médium radio [34]. Le médium sans fil est half-duplex, ce qui signifie qu'un nœud ne peut pas simultanément transmettre et réceptionner des données. La coordination entre les différents nœuds est primordiale pour éviter les transmissions simultanées brouillant le signal radio, ce qui génère des collisions. De plus, le médium est à diffusion, ce qui signifie que toute donnée transmise sera reçue par tous les nœuds présents dans le voisinage de l'émetteur (sur-écoute). Ces multiples réceptions se produisent même si le nœud n'est pas le destinataire des données car le filtrage s'effectue au niveau MAC.

Le composant radio est l'élément le plus gourmand en énergie d'un nœud capteur sans fil [35]. La quantité d'énergie utilisée par le composant radio dépend du mode dans lequel il se trouve (veille, transmission ou réception / écoute). Directement chargé de sa gestion, un protocole MAC doit limiter la déperdition énergétique provenant de ce composant. Pour recevoir une donnée lui étant destinée, un nœud doit écouter le médium radio. La consommation énergétique de ce mode varie d'un composant à l'autre et peut être le plus gourmand en énergie.

En fonction de la puce radio, cette écoute du médium peut consommer beaucoup d'énergie, car le mode réception est en général le plus énergivore de la puce. Par exemple, un composant radio CC2420 consomme 19.7 mA à 2.4 GHz [36] (et ce, même si aucune donnée ne transite sur le médium radio) tandis que transmettre à -10dBm ne consomme que 8.5 mA. A l'opposé, un composant radio CC1101 consomme 14.6 mA à 868 MHz [37] et 16.4 mA pour émettre à la même puissance.

Pour limiter cette consommation liée à la couche physique, les protocoles MAC essaient de placer le composant radio en veille le plus souvent possible.

Les problèmes de transmission peuvent également être une forte source de consommation énergétique. La transmission simultanée de deux trames risque de générer une collision et peut déclencher la retransmission des données. Ces transmissions simultanées peuvent être causées par une mauvaise coordination entre les nœuds ou au problème du terminal caché [38]. Le protocole MAC doit donc être robuste face aux problèmes de transmission tout en limitant son impact sur la consommation énergétique.

1.3 PROTOCOLES MAC POUR RÉSEAUX DE CAPTEURS SANS FIL

Pour réduire drastiquement la consommation énergétique des nœuds, la solution principale consiste à placer le composant radio en mode veille le plus souvent possible. Ceci risque cependant de poser un problème de synchronisation. En effet, les données seront perdues si la radio d'un nœud est en veille lorsqu'une transmission lui est adressée. Les protocoles MAC adaptés aux réseaux de capteurs sans fil devront faire en sorte que la source et le destinataire d'une trame soient éveillés durant la même période pour que la transmission soit un succès. Afin d'assurer cette synchronisation entre les nœuds, deux procédés principaux ont émergé : les protocoles synchronisés et les protocoles à préambule.

1.3.1 *Protocoles MAC synchronisés*

Les protocoles MAC synchronisés résolvent le problème de synchronisation en divisant le temps en intervalles pour tous les nœuds du réseau. Ces protocoles peuvent être divisés en deux catégories.

Le première catégorie divise le temps en plusieurs intervalles discrets, appelés *slots*, qui sont répartis entre les nœuds. Cette catégorie se base donc sur le principe de Time Division Multiple Access (TDMA) et requiert une forte synchronisation glo-

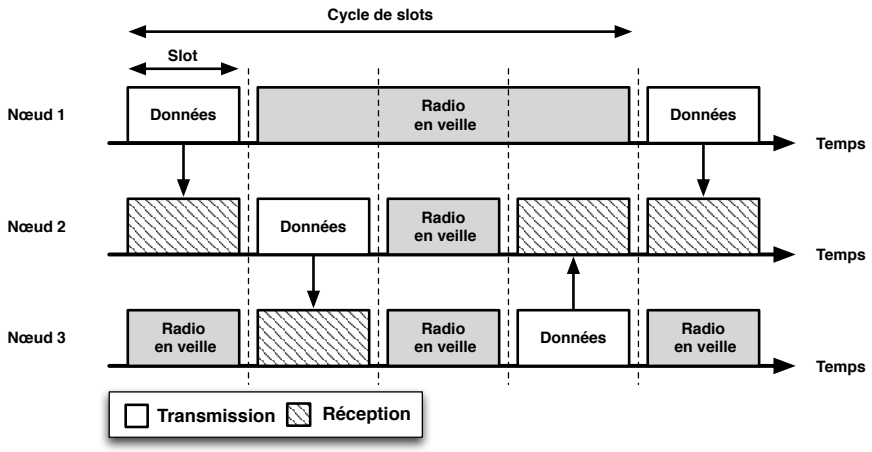


FIGURE 3: Exemple de répartition de slots avec un protocole MAC synchronisé

bale entre tous les nœuds. Un nœud va s'accorder avec ses voisins pour déterminer l'usage qu'il fera des slots qui lui ont été attribués (émission ou réception). Durant le reste du temps, sa radio sera en veille afin d'économiser de l'énergie. La figure 3 illustre cette répartition. Les protocoles MAC TRAMA [39] et TSMP [40] entrent dans cette catégorie.

L'avantage de ces protocoles est l'absence de collision car seul un émetteur est actif durant un slot. À partir d'une topologie donnée, ces protocoles sont capables d'établir une distribution évitant toute collision grâce à une forte synchronisation des nœuds et sont donc adaptés pour des trafics avec des émissions périodiques. Cependant, obtenir la topologie du réseau, répartir les slots et maintenir la synchronisation entre les nœuds va nécessiter de nombreux messages de contrôle ou du matériel coûteux (pour réduire la dérive de l'horloge, par exemple). En conséquence, ces solutions synchronisées visent particulièrement les petits déploiements ayant une topologie fixe.

La seconde catégorie utilise une synchronisation locale et alterne des phases de sommeil et d'activité. Ces phases sont périodiques et le début de chaque phase d'activité est synchronisé localement entre les nœuds du voisinage. Durant la phase de sommeil, la radio est en veille, ce qui permet d'économiser de l'énergie. Durant la phase d'activité, les nœuds accèdent au médium librement. Contrairement aux protocoles de la première catégorie, les nœuds sont en compétition pour transmettre leurs informations et utilisent la méthode CSMA/CA pour accéder au médium. La figure 4 illustre cette alternance de phase. Les

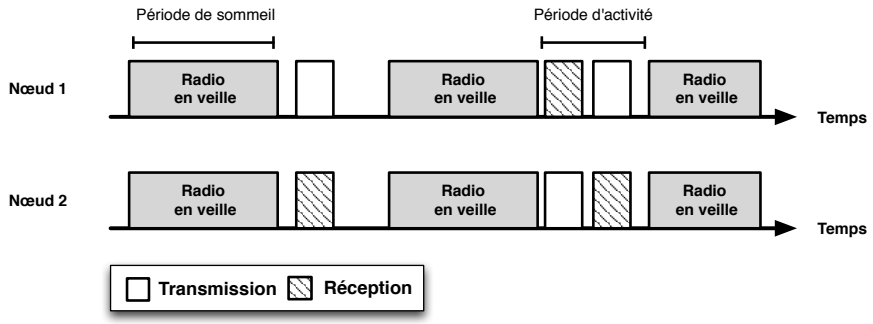


FIGURE 4: Exemple de transmission avec des nœuds partageant des phases d'activité et de sommeil communes.

protocoles S-MAC [41] et T-MAC [42] utilisent ce type de synchronisation. La synchronisation locale de ces protocoles permet de réduire la quantité de messages de contrôle nécessaire mais au détriment de l'assurance de transmissions sans collision. Tout comme celles de la première catégorie, ces solutions visent particulièrement les petits déploiements.

1.3.2 *Protocoles MAC à préambule*

À l'opposé des protocoles synchronisés, les protocoles MAC à préambule permettent aux nœuds de décider par eux-mêmes de l'organisation de leurs périodes d'activité et de sommeil et cela indépendamment de leur voisinage. Dans cette famille, la synchronisation entre l'émetteur et le récepteur n'est pas permanente mais valable uniquement pour la transmission courante. Une nouvelle synchronisation sera nécessaire à chaque nouvelle transmission.

Cette famille est composée de très nombreux protocoles [43]. L'un des premiers protocoles proposés par la communauté scientifique est le protocole Berkeley MAC (B-MAC) [44]. Pour s'assurer que le destinataire est prêt à réceptionner des données, un nœud émetteur va, dans un premier temps, transmettre une série de symboles appelée un préambule. Par défaut, chaque nœud a une durée de sommeil identique. Au début de ce préambule, les nœuds n'étant pas forcément synchronisés, la durée d'émission du préambule est plus longue que la période de sommeil. L'émetteur est ainsi assuré que tous les nœuds dans son voisinage, notamment le destinataire, sont réveillés et prêts à recevoir les données. Un mot de synchronisation est envoyé

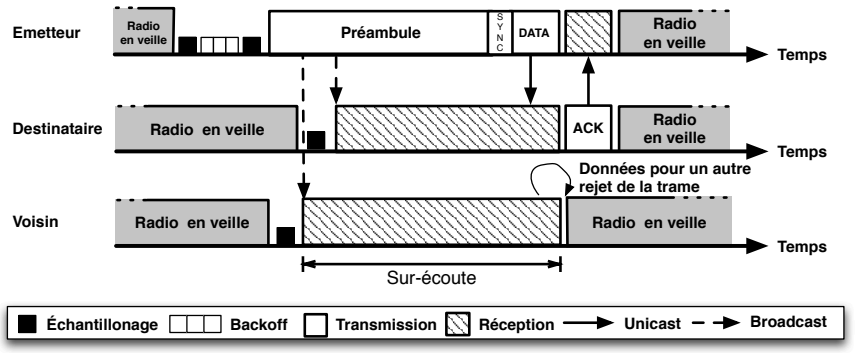


FIGURE 5: Transmission de données avec le protocole B-MAC. Un acquittement est envoyé pour informé l'émetteur de la bonne réception des données.

pour indiquer la fin du préambule et l'imminence de la transmission de la trame contenant les données. Chaque nœud du réseau va réveiller sa radio à intervalle régulier et écouter le médium radio (échantillonnage) durant une très courte période. Si aucun trafic n'est détecté, le nœud éteint sa radio et entame une nouvelle phase de sommeil. Dans le cas contraire, le nœud reste actif pour réceptionner la trame de données. Suivant les fonctionnalités offertes par le protocole MAC, un acquittement (ACK) peut être transmis par le destinataire de la trame de données pour informer l'émetteur de la bonne réception des données. La figure 5 illustre le fonctionnement du protocole B-MAC. Le fonctionnement asynchrone de B-MAC le rend robuste face aux changements topologiques et lui permet d'être utilisé plus facilement sur de grandes topologies.

Même si le protocole B-MAC permet d'économiser de l'énergie, il a néanmoins un point faible important. En effet, les préambules de B-MAC sont constitués uniquement de bruit et sont transmis continuellement. Les nœuds doivent donc rester éveillés jusqu'à la réception de la trame de données avant d'être en mesure d'identifier le destinataire de la transmission. Cela implique une forte sur-écoute et une consommation énergétique inutile pour les nœuds n'étant pas le destinataire.

Le protocole X-MAC [45] offre une solution pour pallier ce problème, en divisant le préambule en micro-trames qui contiennent l'adresse du destinataire de la trame de données. Lors de la réception d'une micro-trame, un nœud vérifie ainsi s'il est le destinataire de la trame de données et, dans le cas contraire, il éteint directement sa radio sans attendre la trame de données.

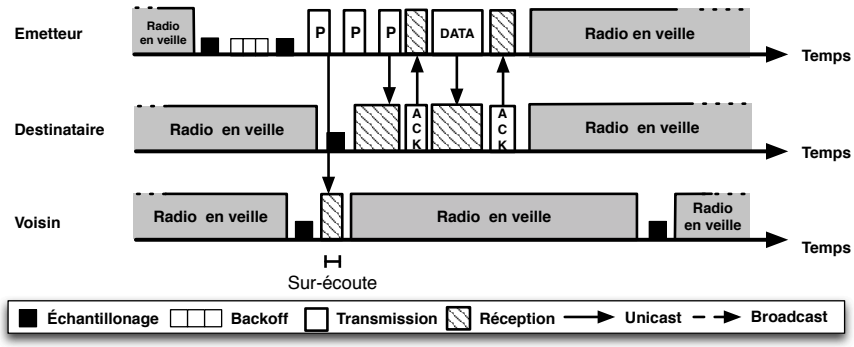


FIGURE 6: Le protocole X-MAC découpe le préambule en micro-trames (P) contenant l'adresse du destinataire.

De plus, pour signifier au plus tôt son réveil, le destinataire acquitte la première micro-trame complète de préambule reçue. Cela permet de débuter immédiatement la transmission de la trame de données (voir figure 6). Acquitter le préambule permet également de réduire le temps d'activité des nœuds et le temps d'occupation du médium radio.

La simplicité de ces protocoles les ont rendus populaires et différentes propositions ont été soumises pour réduire leur consommation énergétique. Dans [46], les auteurs proposent un protocole qui remplace les micro-trames par le paquet de données. Ce remplacement permet d'accélérer la transmission des données : le destinataire devant uniquement acquitter le paquet reçu. Ceci peut cependant poser des problèmes si le paquet de données est volumineux car sa longue durée de transmission le rend plus vulnérable face aux collisions. Le protocole, WiseMAC [47], a pour objectif est de réduire la durée d'émission du préambule. Lors de ses différents échanges avec son voisinage, un nœud peut apprendre à quel moment ses voisins vont échantillonner le médium radio. Il pourra ainsi commencer l'émission du préambule peu de temps avant le réveil du destinataire.

La durée de sommeil des nœuds est l'élément clé pour réduire la consommation d'énergie. Une durée de sommeil trop longue augmentera cependant le délai de bout en bout. Cette durée est définie par l'utilisateur lors du déploiement du réseau mais plusieurs propositions cherchent à l'optimiser une fois le réseau déployé. La solution centralisée, pTunes [48], détermine pour tous les nœuds une valeur optimale de leur durée de sommeil à partir de contraintes applicatives et d'informa-

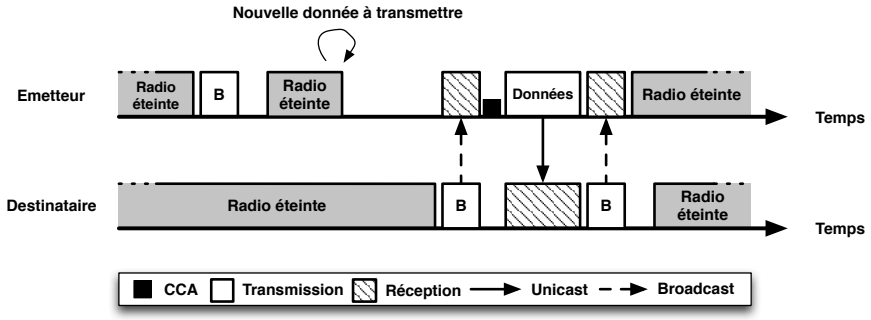


FIGURE 7: Avec RI-MAC, un émetteur va attendre une balise (B) du destinataire avant de transmettre ses données.

tions collectées dans le réseau (délai de bout en bout, énergie restante, etc.). A l'opposé, le protocole BOX-MAC [49] propose d'adapter la durée de sommeil de manière décentralisée. Lorsqu'il n'y a aucun trafic, un nœud aura une longue période de sommeil. A contrario, il diminuera temporairement sa période de sommeil pour réduire le délai de bout en bout, lorsqu'il devra transmettre des données.

Pour avoir un fonctionnement asynchrone, les protocoles de cette catégorie doivent émettre un préambule pour se synchroniser brièvement avec le destinataire de la trame de données. L'émission de ce préambule est cependant coûteuse ce qui rend ces protocoles plus adaptés à des trafics de données irréguliers.

Par ailleurs, pour éviter d'occuper le médium radio avec un préambule, la synchronisation temporaire entre deux nœuds peut être effectuée par le destinataire. Les protocoles utilisant une telle synchronisation sont appelés *orientés récepteurs* et nous pouvons prendre comme exemple les protocoles Low Power Probing [50] ou Receiver-Initiated MAC (RI-MAC) [51]. Ainsi, un nœud utilisant un tel protocole va transmettre une balise à intervalle régulier pour indiquer qu'il est éveillé. Il écoutera donc le médium radio un court instant avant de s'endormir à nouveau s'il ne réceptionne aucune donnée. Ainsi, un nœud souhaitant émettre une trame va simplement allumer sa radio et attendre la réception de la balise correspondant au destinataire du paquet avant de transmettre la trame de données.

En fonction du protocole utilisé, un acquittement peut être envoyé pour confirmer la réception du paquet de données. En cas de collisions entre trames, le protocole RI-MAC propose d'utiliser une fenêtre de contention (dont la taille maximale est

contenue dans la balise) pour éviter toute collision supplémentaire. Le protocole RI-MAC propose également d'utiliser la balise comme acquittement et permet ainsi à d'autres émetteurs de transmettre directement leur paquets. La figure 7 illustre le fonctionnement du protocole RI-MAC.

Ces protocoles peuvent être relativement économes en énergie surtout si la transmission d'informations sur le médium radio est très consommatrice en énergie. Par exemple en fonction de la puissance d'émission, transmettre un paquet peut consommer deux fois plus d'énergie que d'écouter le médium radio avec un composant CC1101. L'inconvénient des protocoles MAC orientés récepteurs est la diffusion d'une trame de données (trame broadcast) car tout le voisinage n'est pas éveillé au moment de la transmission de la trame.

Ce problème peut être résolu de deux manières différentes. La première va utiliser la découverte de voisinage, c'est-à-dire qu'un nœud pourra transmettre en unicast la trame de données aux voisins qu'il connaît. Cette solution peut cependant poser des problèmes lorsque des nœuds sont mobiles car le voisinage est modifié. La seconde solution consiste à transmettre la trame à tous les nœuds émettant une balise durant une période plus longue que la durée de sommeil des nœuds. Il peut cependant y avoir un problème parce qu'un nœud peut réceptionner plusieurs fois la même trame de données.

1.4 IEEE 802.15.4

Ratifié en 2003 par l'organisme de standardisation IEEE, la norme 802.15.4 spécifie la couche physique et la couche MAC pour les réseaux privés sans fil à faible débit. Ces réseaux ont plusieurs points en commun avec les réseaux de capteurs sans fil. Les équipements qui composent ces réseaux ont de faibles débits et de faibles coûts de production. De plus, ils sont alimentés par batterie et disposent donc d'une quantité énergie limitée. La version ratifiée en 2006 complète la définition de la couche MAC et étend les capacités des fréquences existantes (868 MHz, 915 MHz et 2,4 GHz). Des informations supplémentaires sont disponibles dans la section 1.4.2.

Nous allons présenter ci-après les différents acteurs et les différentes topologies introduits par la norme 802.15.4 et nous dé-

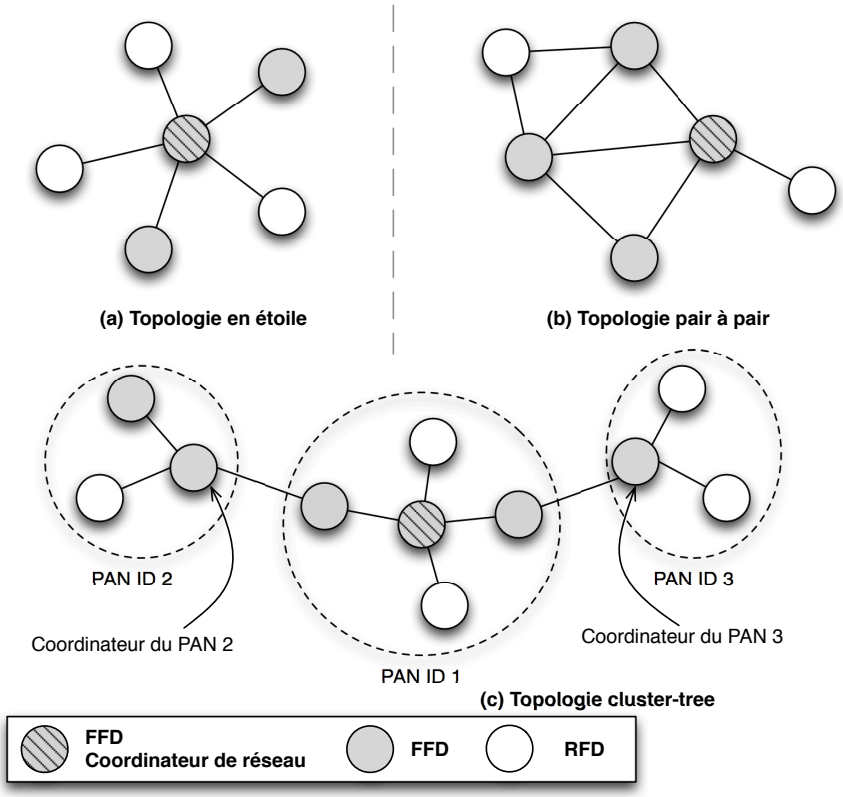


FIGURE 8: Exemple de topologies

taillerons la couche physique et la couche MAC définies par ce protocole.

1.4.1 *Principe de fonctionnement*

Deux principaux types d'acteurs entrent en jeu au sein de la norme 802.15.4 : les *Full Function Device (FFD)* et les *Reduced Function Device (RFD)*. Les RFD disposent de ressources modestes et implémentent une version minimale de la norme. Un RFD est conçu pour des applications simples telles qu'activer ou non une lumière ou récupérer de manière passive la température d'une zone. En raison de leurs limitations, les RFD peuvent uniquement communiquer avec des FFD et sont, par conséquent, les équipements terminaux du réseau. A l'opposé, les FFD sont les équipements principaux du réseau et implémentent une version complète de la norme. Les capacités matérielles d'un FFD lui permettent de créer un réseau auquel d'autres équipements (FFD ou RFD) pourront s'associer. Un

Fréquence (MHz)	Canaux Largeur	Débit (Ko/s)	Localisation	Modulation
868-868.6	1 canal 600 kHz	20	Europe	BPSK
		250		ASK
		100		O-QPSK
902-928	10 canaux 2 MHz	40	USA Canada	BPSK
		250		ASK
		250		O-QPSK
2400-2483	16 canaux 6 MHz	250	Monde	O-QPSK

TABLE 2: Table de fréquence et de bande passante de IEEE 802.15.4.

FFD initiant le réseau en deviendra l'équipement central et est nommé coordinateur de réseau (*Personal Area Network (PAN) Coordinator*). Le coordinateur joue le même rôle que le puits dans les réseaux de capteurs sans fil. De plus, pour permettre aux équipements de les différencier, chaque réseau dispose d'un identifiant unique dans le voisinage.

Selon les besoins des applications, ces deux acteurs peuvent être déployés selon différentes topologies. Dans la topologie en étoile (voir figure 8), tous les acteurs sont regroupés à un saut autour du coordinateur de réseau et toutes les communications passent par le coordinateur. Au sein de la topologie pair à pair, le coordinateur du réseau peut ne pas être directement accessible à tous les équipements. Les FFD vont donc collaborer pour relayer leurs données et celles des RFD jusqu'au coordinateur de réseau. La dernière topologie proposée par la norme est nommée *cluster tree*. Cette topologie est constituée de groupes d'équipements (*clusters*) reliés entre eux et permet d'étendre la couverture du réseau. Cette topologie est utilisée par les protocole haut-niveau Zigbee [52]. La figure 8 illustre ces différentes topologies.

1.4.2 *Couche physique*

La couche physique est responsable de la transmission et de la réception de données sur le médium radio en utilisant une fréquence définie et un mécanisme de modulation. Dans sa version datant de 2006, la norme 802.15.4 donne accès à trois

bandes de fréquence : 868 MHz, 915 MHz et 2.4 GHz. Au total, 27 canaux sont répartis sur ces trois bandes. Le tableau 2 résume les différentes possibilités offertes par la norme 802.15.4. La couche physique détermine également la taille maximale d'une trame 802.15.4 qui est fixée à 127 octets.

La couche physique doit accomplir différentes tâches en plus de la transmission et réception de données. Elle est naturellement chargée de l'allumage et de la mise en veille de la puce radio en fonction des besoins de la couche MAC. Elle est également chargée de mesurer la puissance du signal reçu et de le comparer avec le niveau du bruit radio ambiant. Ce processus est utilisé par le mécanisme *Clear Channel Assessment (CCA)* pour déterminer si le canal est utilisé ou non. La couche physique fournit également des informations sur la qualité du signal (*Radio Strength Signal Indication (RSSI)* et/ou *Link Quality Indication (LQI)*) qui peuvent être utilisées par les couches supérieures.

1.4.3 *Contrôle d'accès au médium*

La norme 802.15.4 définit également l'accès au médium sans fil. Deux modes de fonctionnement ont été envisagés par l'IEEE pour permettre aux nœuds d'accéder au médium : un mode avec contention et un mode synchronisé.

Mode avec contention

Le mode avec contention est un mode sans économie d'énergie et ressemble fortement au mode Distributed Coordination Function (DCF) de 802.11 [53]. Dans ce mode, chaque nœud garde sa radio allumée et écoute en permanence le canal. Pour s'assurer que le canal est libre, un nœud utilise le mécanisme CSMA/CA pour transmettre une trame et ainsi éviter une collision. Un émetteur est notifié de la bonne réception d'une trame par la réception d'un acquittement. La norme IEEE 802.15.4 prévoit un temps d'attente entre deux messages pour permettre au destinataire de traiter les données reçues (variable en fonction de la longueur de la trame). Une trame d'une longueur inférieure ou égale à 18 octets requiert un court temps d'attente (SIFS) alors que les trames plus longues requièrent un temps plus long (LIFS). Ces différents temps d'attente peuvent être

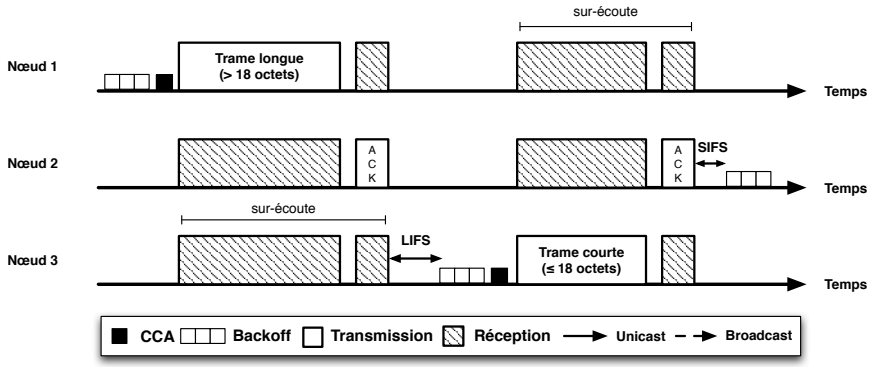


FIGURE 9: Exemple de transmission avec la norme 802.15.4 en mode avec contention.

un inconvénient. En effet, tous les nœuds avoisinants peuvent ne pas entendre la trame de données ainsi que l'acquittement et donc déterminer quand débute et combien de temps dure cet intervalle. La figure 9 illustre la transmission de trames dans ce mode de fonctionnement. Le mode sans contention peut être utilisé sur toutes les topologies présentées dans la section 1.4.1.

Mode synchronisé

Le mode synchronisé de la norme 802.15.4 est une combinaison des deux catégories de protocoles synchronisés présentés dans la section 1.3.1. Les nœuds alternent des phases de sommeil et d'activité communes à un rythme dicté par le coordinateur auquel ils sont associés. La période d'activité est composée d'une supertrame divisée en 16 slots de taille égale. Cette division va permettre aux nœuds de s'assurer des périodes de transmissions sans collision. Pour synchroniser les nœuds, un coordinateur va émettre une balise (*beacon*) durant le premier slot de la supertrame. Ce message de signalisation contient de nombreuses informations sur la supertrame : sa durée, la durée avant l'émission de la prochaine balise, le nombre de slots sans collision, à quels nœuds ils sont réservés, etc. Un exemple de supertrame est représentée dans la figure 10.

Les slots sans collision, nommés *Guaranted Time Slot (GTS)* sont localisés à la fin de la supertrame. Le nombre de GTS et leur répartition sont inclus dans la balise transmise au début de la supertrame. Ainsi chaque nœud dans le voisinage du coordinateur peut s'assurer que ces différents slots sont réservés et ne tentera pas d'émettre dans une période qui ne lui est pas réser-

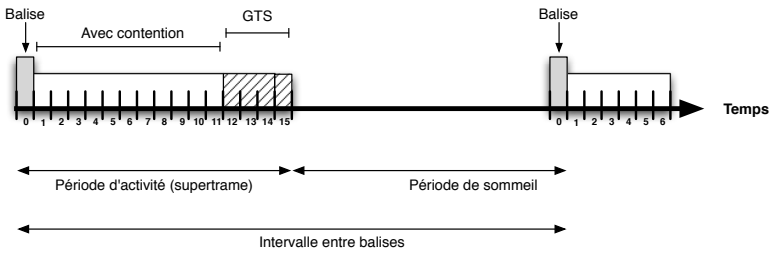


FIGURE 10: Exemple de supertrames IEEE 802.15.4 et leur agencement en cas de topologie cluster-tree

vée. Lorsqu'un slot est réservé, la transmission ne nécessite pas de mécanisme additionnels tel que CSMA/CA.

Les slots non réservés de la supertrame sont utilisables par tous les nœuds. Ces slots peuvent être utilisés pour transmettre des trames de données ou de contrôle (par exemple, pour réserver des GTS). Pour accéder au médium, un nœud doit utiliser un mécanisme de type CSMA/CA. Par rapport au mode avec contention, ce dernier est légèrement modifié pour aligner les CCA sur le début de chaque slot.

Le mode synchronisé est optimisé pour les topologies en étoile ou en *cluster-tree*. Dans une topologie *cluster-tree*, un mécanisme annexe est nécessaire afin d'organiser l'agencement des supertrames. Pour permettre à deux supertrames de cohabiter sur le médium en même temps, ces dernières doivent s'imbriquer. Par exemple dans une topologie *cluster-tree* constituée de deux PAN, la supertrame du second PAN sera inclue dans la période de sommeil du premier PAN. La durée de la supertrame pourra être de longueur inférieure ou égale à la durée de la période de sommeil. Dans un réseau plus complexe, cet agencement de supertrame peut s'avérer difficile.

Les deux principaux avantages du mode synchronisé de la norme 802.15.4 sont respectivement l'économie d'énergie réalisée par les nœuds en mettant en veille leur radio et l'assurance de périodes de transmission sans collision. Cependant, même si ce mode combine le meilleur des deux catégories de protocoles synchronisés (présentées dans la section 1.3.1), ce mode hérite également de leurs défauts. Il est donc difficilement utilisable sur de vastes topologies.

1.4.4 *Adressage des nœuds*

La norme 802.15.4 utilise deux types d'adresses. Le premier type est l'EUI-64 (*64-bits Extended Unique Identifier*) qui est un identifiant codé sur 64 bits considéré comme unique. Cet identifiant est construit à partir de l'adresse MAC sur 48 bits en ajoutant FFFE entre le troisième et le quatrième octet. Tout équipement compatible IEEE 802.15.4 dispose d'une adresse de ce type. Le second type, codé sur 16 bits, est fourni par le coordinateur lorsque qu'un équipement s'associe avec lui. La manière dont ces identifiants sont générés est laissée à l'appréciation de l'utilisateur. En conséquence, ces adresses courtes délivrées par le coordinateur sont valides uniquement dans le PAN ou dans le réseau du coordinateur. L'utilisation d'adresses courtes réduit l'en-tête MAC de 25 octets à 13 octets laissant ainsi plus de place aux couches supérieures.

1.5 GESTION DE LA MOBILITÉ

De nos jours, les réseaux de capteurs sans fil ne sont plus constitués uniquement de capteurs fixes mais également de capteurs placés sur des éléments mobiles. Ces nœuds mobiles peuvent être utilisés dans des applications telles que la surveillance animale [6]. Au niveau MAC, cette mobilité peut se traduire par des difficultés à émettre des données. En effet, le nœud mobile doit s'insérer dans les communications. Cette insertion peut être particulièrement complexe surtout si les communications entre les nœuds fixes sont organisées de manière précise.

Les protocoles MAC synchronisés basés sur des slots vont organiser les communications de chaque nœud de manière précise. Cette forte synchronisation permet d'établir un schéma de communication sans collisions. L'insertion d'un nœud mobile dans les communications impose de renégocier ce programme établi, ce qui peut s'avérer coûteux et complexe. Le protocole MMAC [54] propose de redistribuer à intervalle régulier les slots et d'adapter la longueur des cycles pour permettre aux nœuds mobiles de s'y insérer. La fréquence de redistribution va dépendre du nombre de nœuds mobiles et de leur vitesse. MMAC requiert donc qu'un nœud mobile connaisse sa trajectoire, obtenue à l'aide du système de localisation comme le

système GPS par exemple. La redistribution des slots à intervalles réguliers va générer de nombreux messages de contrôle. Additionnés à ceux nécessaires pour synchroniser l'horloge interne des nœuds [55], ces nombreux messages vont consommer beaucoup d'énergie et réduire la durée de vie des nœuds. Les protocoles MAC synchronisés basés sur des slots ne sont donc pas adaptés pour gérer la mobilité de nœuds dans les réseaux à grande échelle.

Les protocoles MAC basés sur des phases de sommeil et d'activité communes synchronisent localement les nœuds. La phase de synchronisation située au début de la période d'activité est importante car c'est à ce moment que les nœuds sont informés de la durée des différentes périodes. Lorsqu'un nœud se déplace, il peut être complètement désynchronisé par rapport aux nœuds dans son voisinage. Pour pouvoir à nouveau communiquer avec son voisinage, un nœud mobile utilisant le protocole S-MAC [41] garde sa radio allumée jusqu'à la réception d'un message de synchronisation. Cette période d'écoute peut être fortement consommatrice en énergie. Le protocole MS-MAC [24] propose d'augmenter la fréquence des périodes de synchronisation. La fréquence des périodes est adaptée en fonction de la vitesse des nœuds mobiles obtenue à partir du RSSI des messages envoyés. Cependant, le RSSI utilisé pour détecter la mobilité d'un nœud n'est pas un indicateur idéal [56].

Le protocole MAC synchronisé de la norme 802.15.4 requiert que les nœuds soient associés à un coordinateur pour pouvoir échanger des données avec lui. Lors de ses déplacements, un nœud mobile peut se trouver hors de portée du coordinateur et être dans l'impossibilité d'envoyer ou recevoir des données. Pour pallier ce problème, la norme 802.15.4 intègre un mécanisme permettant à un nœud de détecter la perte de son coordinateur. La perte de connexion avec le coordinateur est détectée lorsqu'un nœud ne réceptionne pas quatre balises consécutives. Pour confirmer cette perte, le nœud mobile va envoyer une notification orpheline vers son coordinateur. Sans réponse de la part de son coordinateur après un certain temps, le nœud mobile va chercher à s'associer à un nouveau coordinateur [23]. La norme 802.15.4 est évaluée dans un environnement mobile dans [57]. Cette évaluation a montré que ses performances décroissent en fonction de la vitesse et du nombre de nœuds mobiles.

À cause de leur fonctionnement asynchrone, l'utilisation de protocoles MAC à préambule devient complexe lorsque le réseau devient dynamique. Si un nœud mobile provient d'un réseau différent, il est possible que les durées de préambules diffèrent et rendent impossible toute communication. De plus, la nécessité de transmettre un préambule allonge la durée de transmission, augmentant le risque que le nœud mobile soit hors de portée du destinataire au bout d'un certain temps.

Le protocole X-MACHIAVEL [25] est un protocole MAC basé sur X-MAC [45] dont l'objectif est d'optimiser l'accès au médium des nœuds mobiles. Le protocole X-MACHIAVEL prioritise les communications des nœuds mobiles et leur permet donc de subtiliser le médium radio à un nœud fixe. En pratique, si un nœud mobile souhaite transmettre un paquet de données, mais que le médium est déjà occupé car un préambule est en cours d'émission, le nœud mobile peut transmettre un acquittement spécial. Cet acquittement annonce qu'il subtilise le canal et va transmettre directement son paquet de données au nœud émettant le préambule. De plus, pour accélérer la durée de transmission d'un paquet d'un nœud mobile, tout nœud fixe présent dans le voisinage est autorisé à acquitter le préambule et à devenir la destination du paquet de données. Ce comportement va permettre aux nœuds mobiles d'obtenir plus rapidement l'accès au canal et réduire ainsi les pertes dues à leur éloignement du destinataire.

1.6 CONCLUSION

Dans ce chapitre, nous avons présenté les différentes familles de couche MAC, couche de niveau 2 du modèle OSI. Ces protocoles sont directement responsables de la gestion de la radio et décident donc du moment où elle doit se mettre en veille, transmettre ou écouter le médium. Dans les réseaux de capteurs sans fil, la radio est l'élément le plus énergivore. Pour limiter sa consommation, de nombreux protocoles ont été spécifiquement développés pour ces réseaux avec pour objectif de mettre la radio en veille le plus souvent possible.

Ces nouveaux protocoles offrent différents compromis pour réduire la consommation énergétique en fonction du type d'application visé. Les protocoles synchronisés découpent le temps en intervalles discrets et coordonnent les nœuds afin que ces

Catégorie de protocole	**Nom du protocole**	
		Support de la mobilité
Synchronisés	TRAMA, TSMP, S-MAC, T-MAC, 802.15.4	MMAC MS-MAC, 802.15.4
À préambule	B-MAC, X-MAC, LPP, RI-MAC	X-MACHIAVEL

TABLE 3: Récapitulatifs des protocoles MAC présentés dans ce chapitre

derniers partagent des périodes d'activité communes. Ces protocoles sont donc efficaces lorsque le trafic est régulier. A l'opposé, les protocoles à préambules cherchent à synchroniser les nœuds uniquement lorsqu'une transmission est nécessaire et sont ainsi plus adaptés lorsque le trafic est irrégulier. La table 3 récapitule les différentes catégories de protocoles MAC présentées dans ce chapitre.

La compréhension des protocoles MAC est une première étape dans la gestion de la mobilité. En effet, pour pouvoir communiquer quelque soit sa localisation, un nœud mobile doit pouvoir échanger des données sur le médium radio et être compris par les nœuds avoisinants. Cependant pour qu'un nœud puisse communiquer, quelque soit le réseau dans lequel il se trouve, le nœud mobile et le réseau doivent utiliser le même protocole MAC. Pour s'assurer d'une connectivité totale au niveau 1 et 2, l'organisme de standardisation IEEE a défini une norme, 802.15.4, suffisamment générique pour être utilisée par un maximum d'applications.

La couche 2 permet de communiquer avec des nœuds se trouvant dans le même réseau. Cependant, la communication avec des équipements situés à l'extérieur du réseau devient de plus en plus importante pour faciliter la collecte de données et le contrôle à distance des nœuds. Pour simplifier l'interconnexion entre les réseaux de capteurs sans fil et les autres réseaux, l'utilisation de la technologie IPv6 a été proposée.

2

IPV6 POUR RÉSEAUX DE CAPTEURS SANS FIL

Sommaire

2.1 INTRODUCTION

A l'aide de la norme IEEE 802.15.4, un nœud mobile est capable de communiquer au niveau 2 lorsqu'il se déplace à travers de multiples réseaux sans fil. Cependant, sans protocole adéquat, les communications d'un nœud mobile sont limitées au réseau dans lequel il se trouve. Le protocole IPv6 est utilisé pour interconnecter les équipements à Internet. Le support d'IPv6 permettrait aux réseaux de capteurs sans fil de communiquer directement avec tout équipement connecté à un réseau IPv6.

Après la présentation du protocole IPv6, nous mettrons en avant les différentes limitations liées à son utilisation au sein de réseaux de capteurs sans fil. Nous détaillerons ensuite les adaptations standardisées par l'Internet Engineering Task Force (IETF) [58] pour utiliser IPv6 dans les réseaux de capteurs sans fil en tenant compte de leurs contraintes. De cette standardisation sont nés les réseaux sans fil personnels à basse consom-

mation IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN) [27].

2.1.1 *Internet Protocol version 6*

Le protocole IPv6 [59] est le successeur du protocole IPv4 [60] actuellement utilisé sur Internet. Ce nouveau protocole a été standardisé avec pour objectif : de pallier la pénurie d'adresse (le nombre d'adresse IPv4 disponibles est proche de zéro) et d'améliorer le support des extensions en anticipant sur les besoins futurs tels que la mobilité ou la sécurité.

Différence entre IPv4 et IPv6

Avec la croissance actuelle d'Internet et la volonté d'y connecter les objets de notre quotidien (l'Internet des Objets), le nombre d'adresses IPv4 disponibles s'épuise très rapidement. Par rapport à IPv4, IPv6 multiplie par 4 la taille des adresses, passant ainsi de 32 bits à 128 bits. Cela permettra d'obtenir 3.4×10^{28} adresses et de s'affranchir ainsi du mécanisme NAT qui regroupe derrière une adresse IP publique plusieurs équipements mais de ce fait complexifie le fonctionnement d'applications [61]. Ces nouvelles adresses utilisent un mécanisme d'auto-configuration sans état. L'auto-configuration permet à un équipement de devenir complètement "plug-and-play" et va nécessiter la mise en œuvre de nouveaux protocoles : Neighbor Discovery et ICMPv6. Le passage au protocole IPv6 a permis de simplifier le format de l'en-tête IPv6 (retrait de champs qui ne sont plus utilisés). Il dispose désormais d'une taille fixe : les options d'IPv4 sont devenues des extensions pour IPv6.

Déploiement d'IPv6

Le déploiement d'IPv6 se fait progressivement depuis plusieurs années en raison de la taille du parc de routeurs dans le cœur d'Internet. Pour que les nouveaux réseaux IPv6 puissent fonctionner en attendant la mise à jour des routeurs, des solutions telles que les tunnels IPv6-dans-IPv4 ou l'utilisation d'une double pile IP ont fait leur apparition.

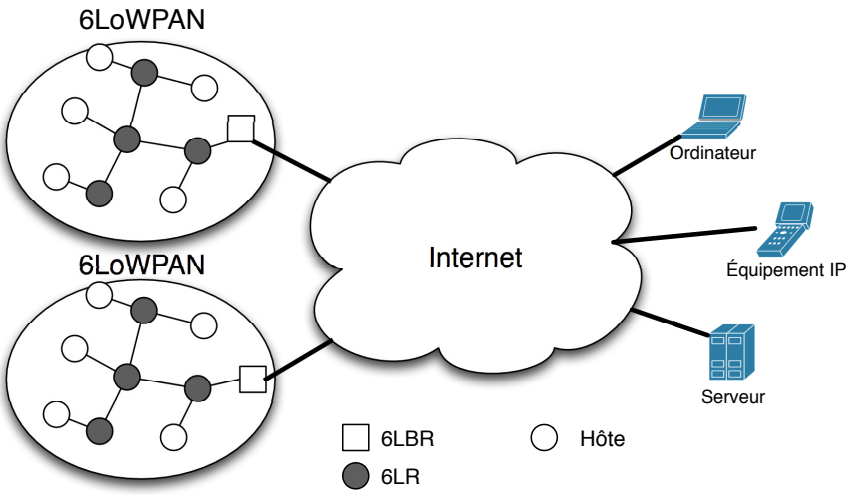


FIGURE 11: Architecture d'un réseau 6LoWPAN

2.1.2 *Problème soulevé*

La taille maximale d'une trame contenant d'un paquet IPv6 est de 1280 octets. En général, au sein d'un réseau de capteurs sans fil, la taille maximale des trames qu'il est possible de transmettre dans ces réseaux est bien inférieure à 1280 octets (la norme IEEE 802.15.4 définit une taille maximale de trame de 127 octets [23]). La surcharge imposée par les en-têtes multiples limitera l'espace disponible pour les données applicatives. Par exemple, une trame de données est composée au minimum des en-têtes MAC, IPv6 et UDP. Ainsi avec la norme IEEE 802.15.4, 25 octets sont utilisés par l'en-tête MAC laissant seulement 102 octets pour les en-têtes de niveau supérieur et les données applicatives. En ajoutant les 40 octets requis par l'en-tête IPv6 et les 8 octets de l'entête UDP, la trame résultante ne laisse que 54 octets pour les données applicatives. L'IETF propose une solution à ce problème en introduisant les réseaux sans fil personnels à basse consommation, 6LoWPAN.

2.2 LES RÉSEAUX 6LOWPAN

Un réseau 6LoWPAN [27] est un réseau de communication simple, à bas coût, permettant d'avoir une connectivité sans fil utilisant une adaptation du protocole IPv6. Il est formé par des équipements, en général compatibles avec le standard IEEE 802.15.4, qui sont caractérisés par une courte portée, un faible

débit, peu de mémoire et un faible coût. Contrairement aux réseaux IP standards, un réseau 6LoWPAN est organisé comme un arbre. Son architecture est présentée dans la figure 11. Il est possible de distinguer trois acteurs dans un réseau 6LoWPAN : le routeur de bordure, le routeur et l'hôte. Le routeur de bordure (ou 6LoWPAN Border Router (6LBR)) est la racine d'un réseau 6LoWPAN. Il connecte le réseau au reste de l'Internet et est responsable de la propagation des préfixes IPv6 au sein du réseau 6LoWPAN. Les routeurs (ou 6LoWPAN Router (6LR)) sont des équipements intermédiaires dont le but est d'étendre la superficie du réseau. Finalement, les hôtes sont les équipements feuilles du réseau. Un réseau 6LoWPAN peut supporter deux types de protocole de routage : *route-over* et *mesh-under*. Plus d'informations sur ces protocoles de routage sont présentés dans la section 2.4.

Afin de permettre l'utilisation IPv6 tout en préservant de l'espace pour les données applicatives, l'IETF a défini une couche d'adaptation. Cette couche d'adaptation a pour objectif de réduire la surcharge de l'en-tête IPv6 et se place entre la couche MAC et la couche réseau.

2.3 COUCHE D'ADAPTATION DE 6LOWPAN

La couche d'adaptation de 6LoWPAN est chargée de réduire la taille des en-têtes en utilisant la compression. Elle propose également des en-têtes supplémentaires lui permettant de fragmenter un paquet IPv6 si ce dernier ne peut pas contenu dans une trame MAC.

2.3.1 *Compression de l'en-tête IPv6*

Un premier mécanisme de compression nommé HC1 a été proposé [27]. Pour réduire la taille de l'en-tête IPv6, HC1 omet des champs spécifiques de l'en-tête IPv6 car ils sont considérés comme implicites (par exemple le champ version a toujours pour valeur 6). En rompant la règle principale du modèle OSI, qui implique que chaque couche doit être indépendante des autres, HC1 peut réutiliser les informations des autres couches (par exemple la longueur peut être calculée à partir de l'en-tête

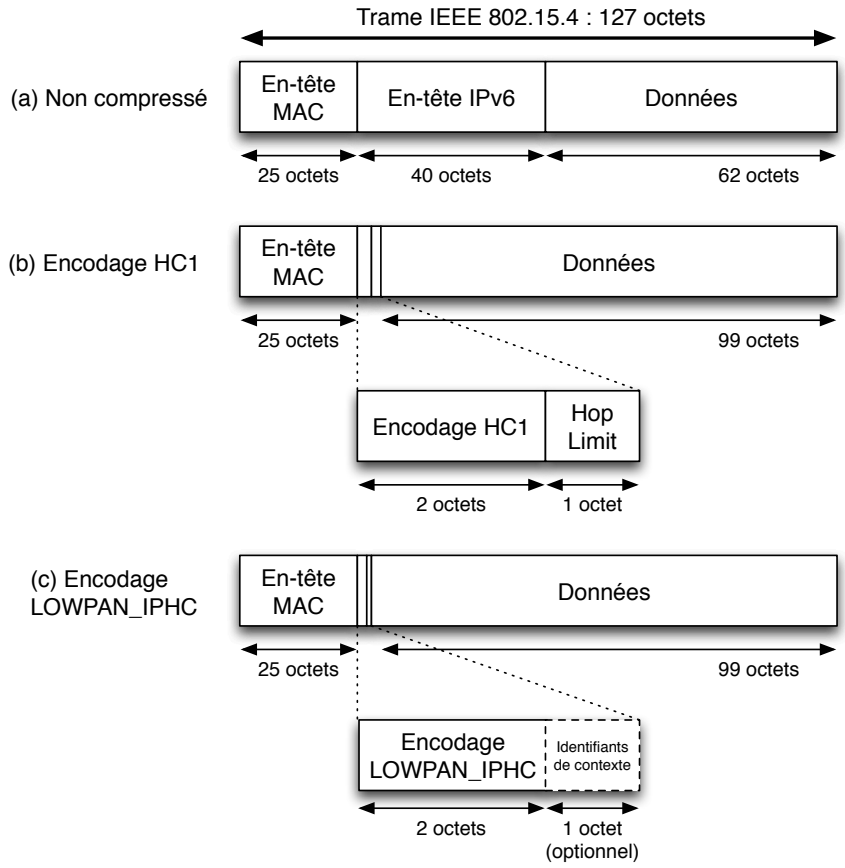


FIGURE 12: Compression d'en-tête avec 6LoWPAN

MAC). Seul le champ *hop limit* de l'en-tête IPv6 reste toujours non compressé.

L'encodage HC1 utilise un pseudo en-tête de 2 octets : un octet pour déterminer le contenu suivant, le champ *dispatch*, (en-tête IPv6, en-tête de fragmentation, etc.) et un second octet pour décrire comment chacun des champs de l'en-tête IPv6 est compressé. La figure 13 présente le format de l'encodage HC1. Un troisième octet, placé après le champ d'encodage HC1, contient la valeur non compressée du champ *hop limit*. Ce mécanisme de compression est donc capable de réduire l'en-tête IPv6 de 40 octets à 3 octets. La figure 12 (b) compare une trame compressée à l'aide de ce mécanisme avec une trame IPv6.

Adressage

Une adresse IPv6 a une taille de 128 bits et est composée d'un préfixe IPv6 (en général sur 64 bits) et de l'identifiant unique

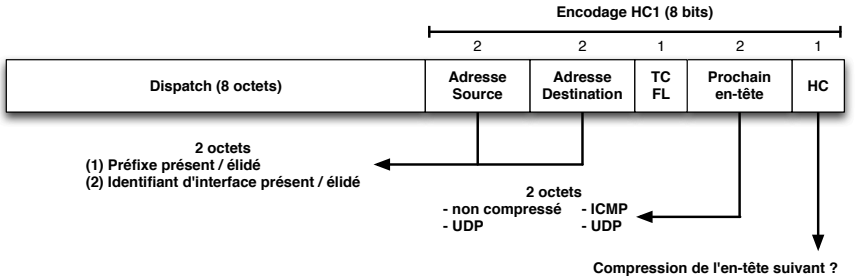


FIGURE 13: Format de l'encodage HC1

étendu sur 64 bits (64-bits Extended Unique Identifier (EUI-64)) de l'interface réseau. L'EUI-64 peut facilement être récupéré à partir de l'en-tête MAC qui est tout simplement le format d'adresse utilisé par la technologie IEEE 802.15.4. De plus, la norme IEEE 802.15.4 définit des adresses courtes sur 16 bits. Si une adresse courte est utilisée, l'identifiant réseau de l'adresse IPv6 est construit de cette manière : `0000:00FF:FE00:XXXX` où `XXXX` est l'adresse MAC courte. Ainsi, l'EUI-64 peut systématiquement être omis lorsque la source ou la destination de niveau 3 correspond à celle de niveau 2.

Le préfixe IPv6 peut être soit le préfixe lien local (`FE80::`) ou un préfixe IPv6 global. L'encodage HC1 peut uniquement compresser des préfixes de type lien local et donc les communications globales incluront systématiquement le préfixe IPv6 global sur 64 bits pour les champs source et destination de l'en-tête IPv6. Cette contrainte limite sérieusement l'utilité de l'encodage HC1, car l'intérêt d'intégrer IPv6 dans les réseaux de capteurs sans fil est permettre des communications globales à l'échelle d'Internet. En conséquence, l'IETF a défini un nouveau mécanisme de compression appelé LOWPAN_IPHC [28].

Un mécanisme de compression plus efficace

Le mécanisme LOWPAN_IPHC utilise un pseudo en-tête sur deux octets. Pour gagner de la place, seulement 3 bits sont utilisés par le champ dispatch. Les 13 bits suivants indiquent pour chaque champ de l'en-tête IPv6, s'il est supprimé ou inclus tel quel. Même si le concept est identique à l'encodage HC1, la manière dont sont compressées les données est différente. La figure 14 présente le nouveau format utilisé. Ce mécanisme permet de compresser le champ *hop limit* en limitant les valeurs à 1, 64 et 128. Cela est particulièrement utile pour des messages de

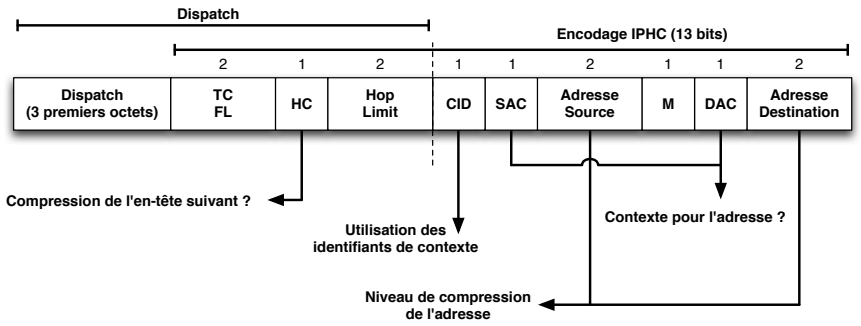


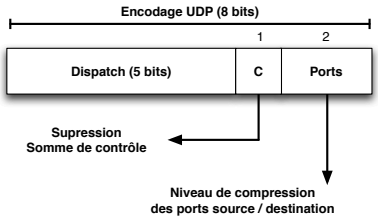
FIGURE 14: Format de l'encodage LOWPAN_IPHC

contrôle valide pour un saut. Il distingue également les champs label (FL) et classe (TC) permettant une compression plus fine.

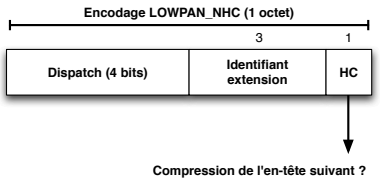
La compression des adresses IPv6 est effectuée en fonction de deux modes : une compression simple ou une compression basée sur les contextes. La compression simple est similaire à la compression d'adresse définie dans HC1. La compression basée sur les contextes est définie pour compresser les adresses IPv6 globales. Pour cela, un identifiant de contexte (codé sur 4 bits) est défini pour chaque préfixe IPv6 global utilisé dans le réseau 6LoWPAN (voir section 2.5 pour plus d'informations sur l'obtention de ces identifiants). Pour les communications globales, le préfixe IPv6 sur 64 bits est remplacé par l'identifiant de contexte correspondant. Dans une telle configuration, un octet supplémentaire est placé juste après le pseudo en-tête LOWPAN_IPHC. Il contient les identifiants de la source et de la destination. Les 64 derniers bits de l'adresse IPv6 sont compressés de la même manière qu'avec l'encodage HC1. Au final, l'encodage LOWPAN_IPHC réduit au maximum l'en-tête IPv6 à 2 octets avec le mode de compression simple et à 3 octets avec une compression basée sur les contextes. La figure 12 (c) présente l'encodage LOWPAN_IPHC.

2.3.2 *Compression d'autres en-têtes*

La couche d'adaptation proposée par 6LoWPAN propose également de compresser l'en-tête UDP. Il permet de réduire la taille cet en-tête de 8 octets à 2 octets au maximum. Il supprime obligatoirement le champ longueur et donne la possibilité de supprimer la somme de contrôle. La figure 2.15(a) illustre le format d'un en-tête UDP compressé. Initialement codés sur 2



(a) Format de l'encodage UDP



(b) Format de l'encodage LOWPAN_NHC

octets chacun, les ports source et destination sont compressés en réduisant la plage de valeurs disponibles selon un intervalle prédéfini. Avec une compression maximale, le numéro du port est codé sur 4 bits autorisant l'utilisation de 16 ports (de 61616 à 61631). Avec une compression moyenne, le numéro du port est codé 8 bits et il est donc inclus dans une plage de 256 ports (de 61440 à 61695).

Le protocole IPv6 fait usage d'extensions afin d'étendre ses capacités. Ainsi, un datagramme UDP ne devient pas forcément un paquet IPv6 contenant un en-tête IPv6 suivi directement de l'en-tête UDP. Pour compresser efficacement un paquet IPv6, la couche d'adaptation spécifie que tous les en-têtes compressés doivent obligatoirement être contigus. Ainsi, pour bénéficier de la compression de l'en-tête UDP tout en utilisant des extensions IPv6, un nouvel encodage LOWPAN_NHC est introduit. L'encodage LOWPAN_NHC est composé du pseudo en-tête LOWPAN_NHC suivit directement par l'extensions IPv6. Le format de ce pseudo en-tête est présenté dans la figure 2.15(b). Il va permettre l'utilisation de ces différentes extensions : saut-à-saut, routage, fragmentation, destination et mobilité mais également d'utiliser un en-tête IPv6 non compressé si besoin. Dans tous les cas, le champ longueur est supprimé de l'extension. De plus, si le bit HC (indiquant la compression du prochain en-tête) est mis à 1, le champ *next header* est également supprimé.

2.4 ROUTAGE

Pour acheminer les paquets vers leur destination, les réseaux 6LoWPAN utilisent un protocole de routage. À la différence des réseaux traditionnels dans lequel le routage est effectué uniquement au niveau 3, il peut être effectué à deux niveaux différents.

2.4.1 *Routage mesh-under*

Dans un premier temps, un réseau 6LoWPAN peut utiliser un routage de niveau 2. Directement intégré à la couche d'adaptation de 6LoWPAN, ce routage, nommé *mesh-under*, met tous les hôtes à un saut IP du 6LBR même si plusieurs saut physiques sont nécessaires. Pour router les paquets sans les décompresser, un en-tête mesh va être ajouté avant le paquet IPv6. Il va contenir l'adresse au niveau 2 du nœud ayant créé le paquet, l'adresse au niveau 2 du destinataire et une limite de sauts (14 au maximum). Ces adresses de niveau 2 peuvent être codées sur 64 ou 16 bits. 6LoWPAN ne définit que l'en-tête mesh. Le calcul des routes ainsi que leur maintenance est effectuée par d'autres mécanismes. Dans les protocoles *mesh-under*, tout nœud peut être susceptible de router un paquet.

Ratifié en 2009 par l'IETF, le protocole 802.15.5 [62] est un protocole de routage *mesh-under* libre. Ce protocole utilise un arbre d'adressage pour router les données sans avoir besoin de table de routage. Initialement, la racine dispose d'une série d'adresses. Elle va ensuite répartir ces adresses de manière contigüe à ses fils en fonction du nombre de fils de chacun, et ainsi de suite. Pour router les paquets, il suffit à un nœud de comparer l'adresse du destinataire avec les adresses dont il dispose. Si l'adresse n'en fait pas partie, il transfère le paquet à son père. Dans le cas contraire, il transfère le paquet vers son fils disposant de la plage d'adresse contenant celle du destinataire.

Si le paquet compressé est trop grand pour être contenu dans une trame, il sera fragmenté en plusieurs morceaux. Cette fragmentation est effectuée par la couche d'adaptation du nœud créant le paquet IPv6. Dans un routage *mesh-under*, seul le destinataire du paquet IPv6 (ou le 6LBR) le reconstruit. La perte de l'un des fragments est l'inconvénient de ce mécanisme car il sera impossible au destinataire de reconstruire le paquet IPv6.

2.4.2 *Routage route-over*

Le routage peut être effectué au sein de la couche réseau (niveau 3). Ces protocoles de routage, de type *route-over*, génèrent un nombre de sauts IP équivalent au nombre de transmissions

physiques requises. Contrairement aux protocoles de routage *mesh-under*, seul les 6LR ou le 6LBR sont autorisés à router des paquets. La présence de 6LR est donc nécessaire pour étendre la zone couverte par le réseau 6LoWPAN. Comme le routage est effectué au dessus de la couche d'adaptation, les paquets fragmentés sont reconstitués à chaque saut, ce qui rend ces protocoles plus robustes face aux pertes de fragments. Nous pouvons citer comme exemple le protocole RPL [63] défini par le groupe de travail *Routing Over Low power and Lossy networks* de l'IETF. Ce protocole de routage utilise des vecteurs de distance pour construire un graphe orienté acyclique et s'en servir pour router les paquets au sein du réseau.

2.5 NEIGHBOR DISCOVERY POUR RÉSEAUX 6LOWPAN

Opérant au niveau 3, le protocole Neighbor Discovery [64] fournit plusieurs mécanismes utilisés par IPv6 : l'auto-configuration d'adresses, la découverte de routeurs, la détection de l'inaccessibilité de voisins et la résolution d'adresses de niveau 3 en adresses de niveau 2. Cependant, ce protocole est difficilement utilisable sur les réseaux 6LoWPAN.

Tout d'abord, Neighbor Discovery fait un usage intensif des communications multicast, comme par exemple lors de la résolution d'adresses. Dans un réseau de capteurs sans fil, les communications multicast nécessitent que chaque nœud capteur sans fil retransmette le paquet car il n'y a pas de mécanisme de diffusion global au niveau réseau. Par conséquent, les communications multicast génèrent potentiellement un grand nombre d'émissions et impactent sérieusement sur les réserves énergétiques des nœuds. Par ailleurs, Neighbor Discovery suppose que les nœuds soient accessibles tout le temps. Hors, pour réduire la consommation énergétique, les nœuds éteignent leur radio et ne peuvent donc pas être atteints à tout moment. Neighbor Discovery n'est également pas conçu pour des liens sans fil non bidirectionnels. Finalement, au sein de réseaux 6LoWPAN, il est possible que deux nœuds puissent communiquer au niveau 2 sans appartenir au même réseau IPv6. En conséquence, le groupe de travail 6LoWPAN a proposé une adaptation du protocole Neighbor Discovery pour réseaux 6LoWPAN [65].

Obtention d'un préfixe

La première étape a été de réduire les communications multicast en rendant optionnel l'envoi périodique des messages *router advertisement*. De ce fait, un nœud doit nécessairement envoyer un message *router solicitation* pour obtenir et actualiser les informations sur les préfixes IPv6. Lorsqu'un nœud arrive dans le réseau (dû à son démarrage ou à sa mobilité), il envoie un premier message *router solicitation* en multicast. Dès que que ce dernier dispose de l'adresse d'un routeur, il enverra les prochains messages *router solicitation* en unicast. À la réception d'un message *router solicitation*, un 6LBR ou un 6LR envoie en réponse un message *router advertisement* en unicast au nœud effectuant la demande. Les messages *router advertisement* contiennent toujours l'option *source link layer address (SLLA)* (utilisant l'EUI-64 du nœud) et incluent deux nouvelles options :

- *authoritative border router* : cette option contient l'adresse IPv6 du 6LBR (pratique pour un routage de type *route-over*),
- *6LoWPAN context option* : cette option contient l'identifiant de contexte pour un préfixe spécifique.

Après avoir reçu un message *router advertisement*, le nœud construit son adresse IPv6 globale à partir de l'un des préfixes inclus dans le message et de son adresse MAC. L'adresse MAC peut être l'EUI-64 ou l'adresse MAC courte qui sera transformée comme présentée dans la section 2.3.1.

Enregistrement de l'adresse IPv6 globale

L'adaptation du protocole Neighbor Discovery définit un nouveau mécanisme remplacement du mécanisme de résolution d'adresse qui utilisait un grand nombre de messages envoyés en multicast. Désormais, les routeurs maintiennent un cache de voisinage contenant toutes les adresses IPv6 de ses nœuds fils.

Pour maintenir ce cache de voisinage, le nœud va s'enregistrer auprès de son routeur. Pour cela, le nœud envoie un message *neighbor solicitation* qui contient l'option SLLA et une nouvelle option nommée *address registration*. Lors de la réception, le routeur s'assure que l'adresse IPv6 globale du nœud est unique dans le réseau en utilisant son cache de voisinage. Si l'adresse est disponible, le routeur enregistre le couple (adresse MAC,

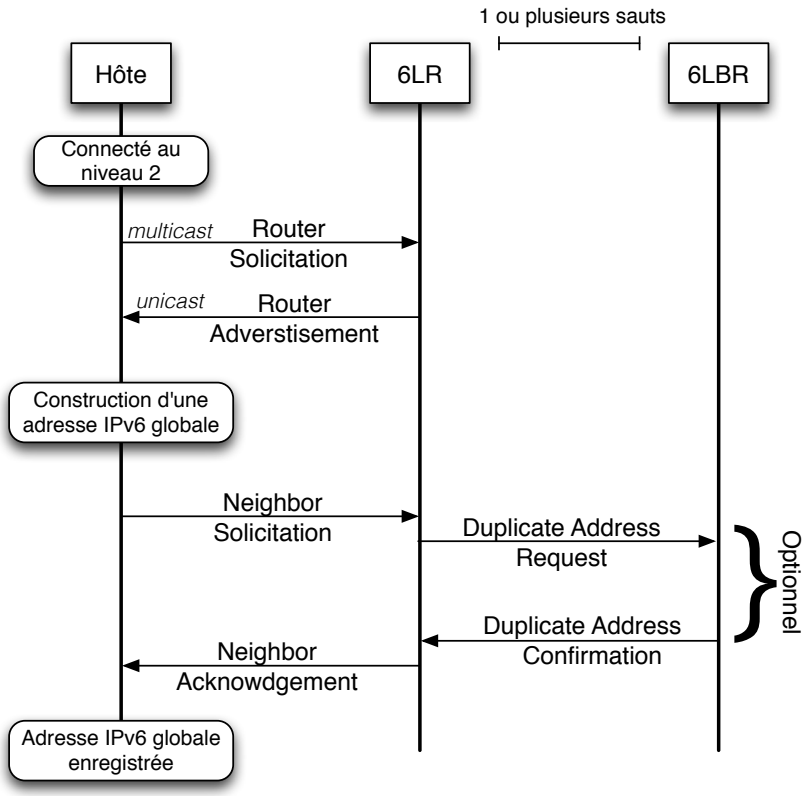


FIGURE 15: Échanges du protocole Neighbor Discovery dans le cadre d'un routage route-over

adresse IPv6 globale) dans le cache de voisinage et répond par un message *neighbor advertisement* qui contient l'option *address registration* avec le champ statut complété. Si l'enregistrement a réussi, le nœud peut commencer à utiliser l'adresse IPv6 globale configurée. Les nœuds envoient périodiquement des messages *neighbor solicitation* pour maintenir le routeur informé de leur présence. Ils envoient également des messages *router solicitation* périodiques pour mettre à jour les autres informations fournies par le routeur. La figure 15 illustre la configuration et l'enregistrement d'une adresse IPv6 globale dans un réseau 6LoWPAN.

Vérification de l'unicité de l'adresse IPv6 globale

L'EUI-64 est considéré comme unique et donc une adresse IPv6 générée à l'aide de cet identifiant est également considérée comme unique. Hors, une adresse MAC courte peut ne pas être unique au sein du réseau, il y a uniquement 2^{16} possibilités. Si

le nœud décide d'utiliser son adresse MAC courte, une étape supplémentaire est requise lors de l'enregistrement de l'adresse pour vérifier son unicité dans le cadre d'un routage *route-over*.

Lorsqu'un 6LR réceptionne un message *neighbor solicitation* utilisant une telle adresse, il va vérifier l'unicité de cette dernière auprès du 6LBR. Pour cela il va lui envoyer un nouveau message ICMPv6, nommé *duplicate address request*. Ce message contient l'EUI-64, l'adresse IPv6 globale du nœud ainsi que sa durée de vie. À la réception d'un tel message, le 6LBR va vérifier l'unicité de l'adresse en la comparant avec les entrée de la table de duplication. Cette table fonctionne exactement comme le cache de voisinage mais recense uniquement les adresses IPv6 basées sur une adresse MAC courte de tout le réseau. Le 6LBR va ensuite répondre par un message *duplicate address confirmation*. Ce message est identique au premier, le 6LBR complète uniquement le champ statut avec la bonne valeur avant de le retransmettre vers le 6LR. Une fois la réponse obtenue du 6LBR, le 6LR peut finalement envoyer le message *neighbor advertisement* au nœud. Dans le cas où l'enregistrement de l'adresse est refusé, le nœud peut recommencer la procédure en utilisant son EUI-64 pour construire son adresse globale. Cette demande de vérification est illustrée dans la figure 15.

2.6 CONCLUSION

Dans ce chapitre, nous avons présenté le protocole IPv6 qui est utilisé pour interconnecter les équipements à Internet. IPv6 définit de nouveaux mécanismes pour permettre aux différents équipements de configurer automatiquement leur adresse sans utiliser de services d'adressage. L'utilisation d'IPv6 au sein des réseaux de capteurs va permettre aux nœuds d'avoir une connectivité de niveau 3 et leur permettre ainsi de communiquer hors du réseau dans lequel ils se trouvent.

Cependant, l'intégration d'IPv6 pose principalement deux problèmes : le protocole Neighbor Discovery utilise de nombreux messages envoyés en multicast gourmands en énergie et le cumul des différents en-têtes va réduire drastiquement l'espace disponible dans la trame pour les données applicatives. Pour résoudre ces problèmes, les réseaux 6LoWPAN ont été standardisés par l'IETF. Ces réseaux intègrent une couche d'adaptation d'IPv6 et utilisent une version modifiée du protocole Neigh-

bor Discovery utilisé lors de l'auto-configuration sans état de l'adresse IPv6.

L'intégration d'IPv6 est un sérieux avantage pour les nœuds capteurs qui peuvent désormais transmettre leurs données vers leurs correspondants distants via Internet sans nécessiter de protocole de traduction au niveau du 6LBR. Cependant, lors de leur déplacements, un nœud mobile va obtenir des adresses IPv6 globales qui pourront être différentes en fonction du réseau 6LoWPAN traversé. Sans support de la mobilité au niveau 3, un correspondant réinitialisera sa session avec le nœud mobile si le paquet a une adresse source différente de celle attendue. Dans le prochain chapitre, nous nous intéresserons aux protocoles supportant la mobilité au niveau 3 et à leur utilisation dans les réseaux de capteurs sans fil.

3

MOBILITÉ IPV6

Sommaire

3.1 INTRODUCTION

Le protocole IPv6 permet aux nœuds capteurs sans fil de communiquer avec d'autres équipements situés hors du réseau dans lequel ils se trouvent sans requérir de mécanisme intermédiaire effectuant la traduction entre le réseau de capteurs sans fil et Internet. Dans le chapitre précédent, nous avons présenté les réseaux 6LoWPAN qui intègrent une couche d'adaptation permettant d'utiliser efficacement le protocole IPv6 tout en tenant compte des contraintes des réseaux de capteurs sans fil.

Lors de ses déplacements, un nœud mobile peut obtenir différentes adresses IPv6. Sans support spécifique, les différentes connections entre un nœud mobile et ses correspondants seront réinitialisées car ces derniers pourront supposer que les paquets proviennent d'un autre nœud. Pour résoudre ce problème, l'IETF a standardisé différents protocoles supportant la mobilité au niveau 3.

Dans ce chapitre, nous présenterons les différents mécanismes et protocoles standardisés par l'IETF permettant aux nœuds mobiles de maintenir leur connectivité de niveau 3 durant leur

déplacement à travers des réseaux IPv6. Nous étudierons ensuite comment ces protocoles peuvent être utilisés et / ou adaptés aux réseaux de capteurs sans fil.

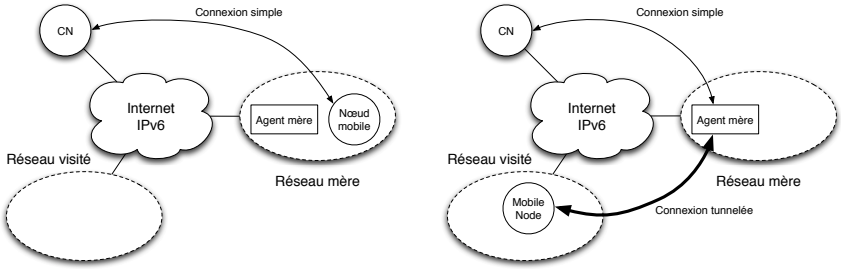
3.2 GESTION DE LA MOBILITÉ

3.2.1 *Problématique*

Sans support spécifique de la mobilité dans le protocole IPv6, les paquets destinés à un nœud mobile ne seraient pas en mesure de lui être délivrés lorsqu'il se déplace hors de son réseau. En effet, les correspondants du nœud mobile transmettent les paquets vers son ancienne adresse devenue obsolète. Pour pouvoir continuer à communiquer malgré ses mouvements, un nœud mobile va obtenir une nouvelle adresse IPv6 dès son arrivée dans le réseau. Néanmoins cela ne sera pas suffisant pour garder une continuité des communications de niveau 3. En réceptionnant un paquet provenant d'une autre adresse, les correspondants vont réinitialiser leur session avec le nœud mobile en pensant que le paquet a été émis par un autre nœud. Pour résoudre ce problème, le protocole Mobile IPv6 a été proposé.

3.2.2 *Le protocole Mobile IPv6*

Mobile IPv6 [29] est un standard IETF dont l'objectif est la gestion de la mobilité IPv6 (niveau 3). L'objectif de ce protocole est de maintenir la connectivité entre un nœud mobile et ses correspondants en utilisant une station relais : l'agent mère. Pour un nœud mobile, les réseaux vont se diviser en deux catégories : un réseau mère, dans lequel le nœud est initialement déployé, et une multitude d'autres réseaux (réseaux visités), dans lesquels le nœud mobile peut se déplacer. Pour maintenir les connexions entre un nœud mobile et ses correspondants, Mobile IPv6 utilise l'agent mère pour rediriger les paquets IPv6 en provenance ou à destination de nœuds mobiles. Cet équipement est installé dans le réseau mère du nœud mobile.



(a) Depuis son réseau mère (b) Depuis un réseau visité

FIGURE 16: Communications avec un nœud mobile à l'aide de Mobile IPv6

Fonctionnement

Lorsqu'il se situe dans son réseau mère, un nœud mobile utilise son adresse mère et communique avec ses pairs en utilisant le protocole IPv6 (voir figure 3.16(a)). Dès qu'il se déplace au sein d'un réseau visité, le nœud mobile va acquérir une nouvelle adresse IPv6 temporaire valide au sein de ce réseau. Le nœud mobile va alors notifier à son agent mère que sa localisation a changé en lui transmettant son adresse temporaire. L'agent mère va alors rediriger le trafic entre le nœud mobile (vers sa nouvelle adresse temporaire) et ses correspondants (CNs) (et vice-versa). Pour cela, un tunnel bidirectionnel IPv6 dans IPv6 est établi entre le nœud mobile et son agent mère. Pour un correspondant, la transmission d'un paquet IPv6 est identique quelque soit la localisation du nœud mobile. Lorsqu'un paquet pour un nœud mobile arrive dans le réseau mère, alors que le nœud mobile est absent, l'agent mère va intercepter le paquet et le retransmettre via le tunnel. L'opération inverse est effectuée lors de la transmission de paquets par le nœud mobile. De ce fait, la mobilité est entièrement transparente pour les correspondants du nœud mobile puisque que ce dernier est toujours accessible depuis son adresse mère. Ce schéma de communication est illustré dans la figure 3.16(b).

Pour maintenir à jour la position de chaque nœud mobile, l'agent mère utilise une table d'association liant l'adresse mère d'un nœud, son adresse temporaire et une durée de vie. Pour cela, le nœud mobile et l'agent mère échangent des messages *binding update* et *binding acknowledgment*. Cet échange est déclenché lorsque le nœud mobile obtient une nouvelle adresse temporaire ou lorsque la durée de vie de son adresse actuelle

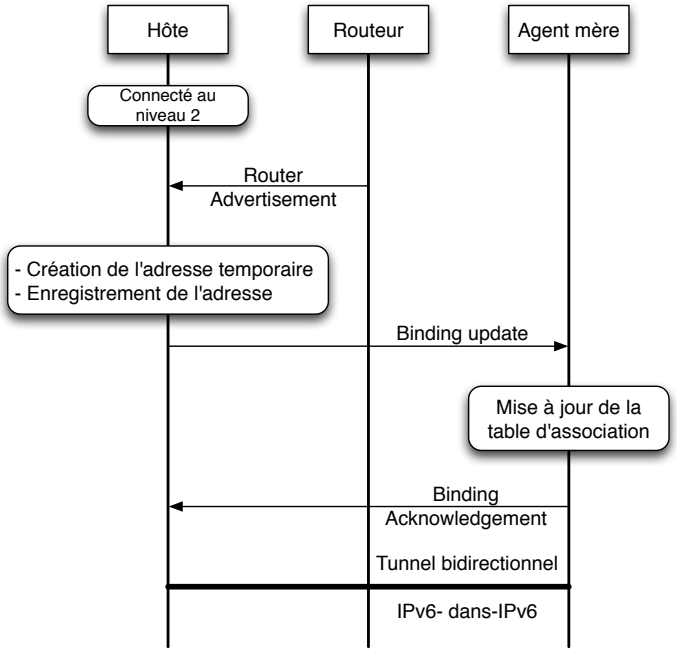


FIGURE 17: Prodécure de handover de Mobile IPv6

arrive à expiration. La figure 17 présente les différents échanges de Mobile IPv6.

La procédure permettant de maintenir les communications durant le passage d'un réseau à un autre, ou *handover*, de Mobile IPv6 se déroule de la manière suivante : lorsque le nœud mobile détecte son arrivée dans un nouveau réseau (par la réception de messages *router advertisement*), il va chercher à obtenir une adresse IPv6 temporaire valide au travers des mécanismes d'auto-configuration. Une fois cette dernière obtenue, il va envoyer un message *binding update* vers son agent mère. La procédure se termine par la réception d'un message *binding acknowledgment*.

Malgré sa popularité, Mobile IPv6 souffre de plusieurs limitations. Lorsque il se déplace d'un réseau à un autre, un nœud mobile peut constater une perte de connexion ou de paquets. Cela est principalement dû au temps requis pour :

1. détecter l'arrivée dans un nouveau réseau IPv6 : basé sur la réception d'un *router advertisement*
2. acquérir une adresse temporaire et vérifier son unicité : à l'aide du mécanisme de duplication d'adresse du protocole Neighbor Discovery

3. enregistrer cette adresse temporaire auprès de l'agent mère qui dépend majoritairement du RTT entre le nœud mobile et l'agent mère.

3.2.3 *Extensions de Mobile IPv6*

Plusieurs extensions ont été proposées par l'IETF pour optimiser les performances du protocole Mobile IPv6.

Fast Handovers for Mobile IPv6

Standardisé en 2005, le protocole Fast Handovers for Mobile IPv6 [30] a pour objectif de réduire le temps de déconnexion au niveau 3 du nœud mobile. Pour cela, un nœud mobile est en mesure de demander des informations à propos des points d'accès environnants. A partir de ces informations, le nœud mobile peut préparer son handover et ainsi transmettre / réceptionner des paquets dès qu'il se reconnecte à un réseau.

HMIPv6

Le protocole Hierarchical Mobile IPv6 [66] va également chercher à augmenter la vitesse du handover. Pour cela, ce protocole utilise une nouvelle entité appelée localisée dans chaque domaine. Cette entité joue le rôle d'agent mère au niveau local et ainsi masquer le trafic de signalisation durant la mobilité au sein de son domaine. De plus pour réduire également le délai d'enregistrement, un mécanisme est inclus pour garder valide l'adresse IPv6 temporaire durant tout le déplacement interne à un domaine du nœud mobile.

3.2.3.1 *Proxy Mobile IPv6*

Proxy Mobile IPv6 [31] utilise une approche différente de Mobile IPv6 pour supporter la mobilité des nœuds mobiles. Contrairement à Mobile IPv6, le déplacement du nœud mobile est totalement transparent pour le correspondant et le nœud mobile. Proxy Mobile IPv6 utilise une architecture différente de Mobile IPv6 avec un nouvel équipement. Cet équipement, appelé passerelle d'accès mobile, va effectuer tout le trafic de

signalisation (basé sur celui de Mobile IPv6) à la place d'un nœud mobile. Pour que cela soit possible, le déplacement des nœuds mobiles est limité à un ensemble de réseaux appartenant à un même domaine Proxy Mobile IPv6.

Network Mobility Basic Support

L'IETF a également envisagé que tout un réseau peut être mobile et a proposé le protocole Network Mobility Basic Support [67]. Ce protocole repend le concept et l'architecture du protocole Mobile IPv6 mais déplace la gestion de la mobilité sur le routeur du réseau mobile. Ce dernier, désormais appelé routeur mobile, est donc chargé de transmettre les différents messages de signalisation au nom du réseau. Le routeur mobile transmet par le biais des messages *router advertisement* un préfixe qui est identique tant que ce dernier reste attaché à l'agent mère. Pour les équipements constituant le réseau mobile, la mobilité est totalement transparente tout comme avec Proxy Mobile IPv6.

3.3 MOBILITÉ DANS LES RÉSEAUX DE CAPTEURS SANS FIL

Au sein de l'Internet des Objets, le déplacement des nœuds capteurs mobiles ne sera pas limité à un seul réseau 6LoWPAN. Pour maintenir la connectivité d'un nœud mobile avec ses correspondants, il est nécessaire d'utiliser l'un des protocoles de support de la mobilité présenté au début de ce chapitre. La simplicité du protocole Mobile IPv6 en fait une solution idéale pour nœuds 6LoWPAN qui sont fortement contraints.

Les différents mécanismes utilisés par Mobile IPv6 s'intègrent parfaitement dans les réseaux 6LoWPAN. Mobile IPv6 définit deux nouveaux types de messages (*binding update* et *binding acknowledgment*) qui sont contenus dans une trame 802.15.4. Le format de ces messages est présenté dans la figure 18. L'en-tête de mobilité de type *binding update* ou *binding acknowledgment* a une taille de 12 octets. De plus, un message *binding update* comporte une option de destination qui inclut l'adresse mère du nœud (sur 20 octets). De manière similaire, un message *binding acknowledgement* inclut un en-tête de routage de type 2 ayant une taille de 12 octets. Ces messages de petite taille peuvent donc être contenus dans une trame 802.15.4 sans devoir être

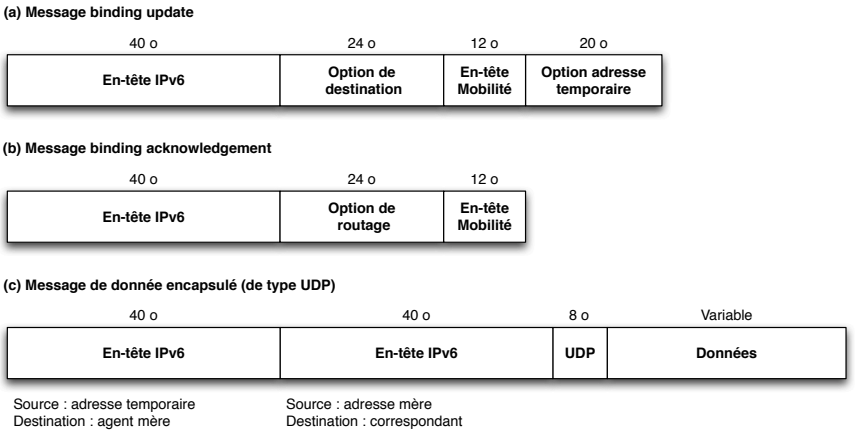


FIGURE 18: Format des messages de mobilité utilisés par Mobile IPv6

fragmentés. Finalement, l'encapsulation IPv6 dans IPv6 utilisée pour le tunnel bidirectionnel exploite des fonctions natives d'IPv6 qui sont toujours disponibles dans la couche d'adaptation de 6LoWPAN. Cette encapsulation va néanmoins laisser moins de place pour les données à transmettre.

3.3.1 *Mécanisme de compression*

Afin de réduire l'espace utilisé par les nouveaux messages de Mobile IPv6, un nouveau mécanisme de compression a été proposé [68]. Basé sur le mécanisme de compression HC1, ce mécanisme va réduire la taille de l'en-tête de mobilité et ses options. Il compresse les champs en changeant le jeu de valeurs disponibles. Par exemple, le numéro de séquence, défini initialement sur 16 bits, est réduit à 5 bits. La dernière méthode de compression change l'unité de certains champs. Par exemple, l'unité de la durée de vie est doublée et définie à 8 secondes comparée à l'unité définie dans [29]. Cette adaptation étant relativement récente, elle ne propose à l'heure actuelle aucune compression pour les options de mobilité (telle que l'option *adresse temporaire alternative*) utilisée par les messages de Mobile IPv6.

Le gain de ce mécanisme de compression dépend de l'en-tête à compresser. Un message *binding update* non compressé utilisé 32 octets : 12 octets pour l'en-tête de mobilité et 20 octets pour les options. Le même message *binding update* utilise plus que 23 octets une fois compressé : 3 octets pour l'en-tête de mobilité et toujours 20 octets pour les options. Les nouveaux formats

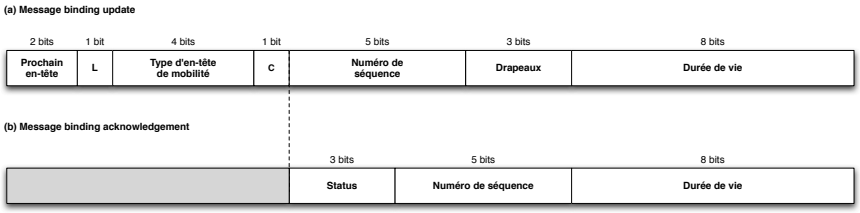


FIGURE 19: Format des messages compressés de Mobile IPv6

de l'en-tête de mobilité sont illustrés dans la figure 19. Avec des ajustements mineurs, le mécanisme de compression devrait être compatible avec l'encodage LOWPAN_IPHC.

3.3.2 *Évaluation du support de la mobilité dans les réseaux 6LoWPAN*

Les évaluations des protocoles de support de la mobilité se divisent en deux catégories : ceux qui évaluent leur utilisation au sein des réseaux de capteurs sans fil et ceux proposant des solutions pour réduire le temps du handover.

Utilisation sur des nœuds capteurs

Dans [69], les auteurs ont comparé les performances d'un nœud mobile utilisant Mobile IPv6 avec un nœud mobile utilisant uniquement IPv6. Durant les expérimentations, le nœud mobile se déplace entre deux réseaux IPv6 : le réseau mère et un réseau visité. Un nœud correspondant, localisé dans un troisième réseau, envoie 100 paquets ICMPv6 vers le nœud mobile. Au $50^{\text{ième}}$ paquet, le nœud mobile passe du réseau mère au réseau visité. Les différents points évalués sont le ratio de livraison, le RTT, l'énergie consommée et l'impact de la fragmentation. Leurs expérimentations ont montré que l'utilisation de Mobile IPv6 triple la consommation énergétique du nœud mobile. De plus, lorsque la fragmentation de paquets est nécessaire, le taux de perte de paquets atteint 80%. A partir de ces résultats, les auteurs ont conclu que Mobile IPv6 n'est pas une solution utilisable pour de la mobilité de niveau 3 dans les réseaux de capteurs sans fil. Cependant, le manque d'informations à propos de nombreux paramètres, tels que l'implémentation et le mécanisme de détection de mouvement utilisé, rend leur expérimentation non reproductible. De plus, cette évalua-

tion n'utilise aucun mécanisme de compression pourtant existant.

Dans [70], les auteurs proposent également une analyse de Mobile IPv6 dans les réseaux 6LoWPAN. Les auteurs se focalisent sur la surcharge des paquets de données et mettent en avant la grande taille des messages de signalisation de Mobile IPv6. Ils observent que la taille des messages *binding update* et *binding acknowledgment* sont similaires à la taille des paquets de données, respectivement 32 et 24 octets. Les auteurs en concluent que Mobile IPv6 n'est pas une solution viable pour la mobilité de niveau 3 dans les réseaux de capteurs sans fil. Cependant, ces messages sont uniquement transmis lorsque le nœud mobile entre dans un nouveau réseau IPv6 ou pour mettre à jour les informations d'association. La taille de ces messages n'a donc aucun un impact significatif sur les performances du nœud mobile.

À partir de ces deux références, la communauté scientifique a conclu que Mobile IPv6 n'est pas utilisable au sein des réseaux 6LoWPAN. Elle a donc recherché de nouvelles solutions pour gérer la mobilité au niveau 3. Parmi ces propositions, nombreuses sont basées sur Proxy Mobile IPv6.

Réduction du temps de handover

Le déploiement de Proxy Mobile IPv6 au sein de réseaux de capteurs sans fil a été évalué dans [71, 72]. Dans [71], les 6LBR vont jouer le rôle de passerelle d'accès mobile et transmettre le trafic de signalisation à la place des nœuds mobiles. Cependant, les auteurs n'ont considéré aucun mécanisme de compression d'en-tête dans leur évaluation (bien que l'encodage HC1 ait été standardisé en septembre 2007). Nous sommes convaincus que les 6LBRs, déjà responsables de l'intégralité du réseau 6LoWPAN (routage, gestion des préfixes IPv6), en plus de faire le pont entre IPv6 adapté et IPv6 classique, ne sont pas capables de supporter également la gestion de la mobilité de nœuds sans impacter leurs performances générales. Dans leur second article [72], les auteurs ont comparé par simulation la durée du handover de Mobile IPv6 et de Proxy Mobile IPv6. Les nœuds implémentent le mécanisme de compression HC1 et une version standard du protocole Neighbor Discovery. Les résultats de leur expérience montrent que la durée du hando-

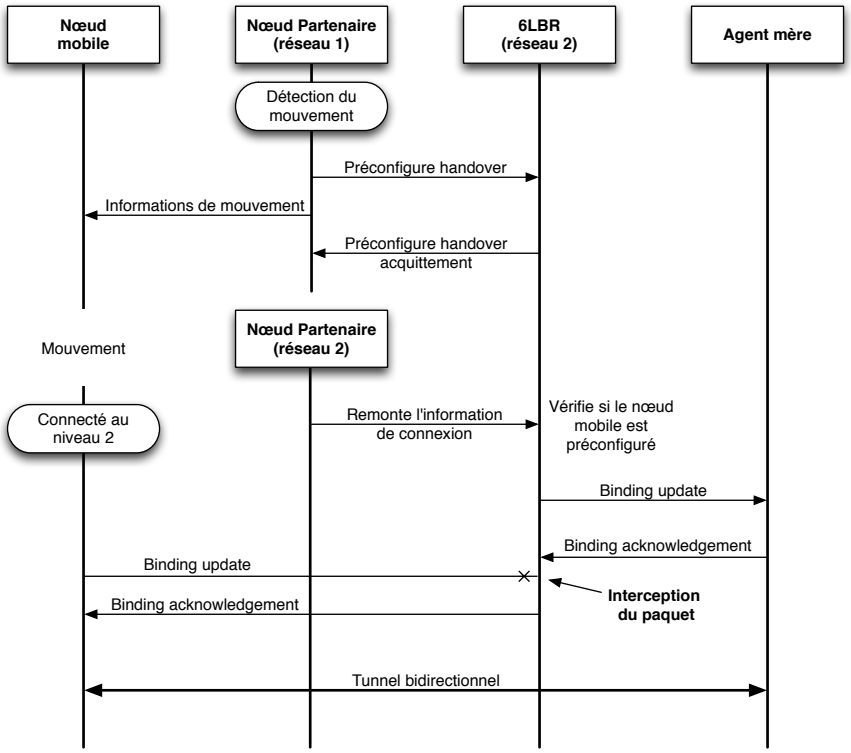


FIGURE 20: Échanges de messages effectués durant un handover avec Inter-MARIO

ver a été divisée approximativement par 10 avec Proxy Mobile IPv6. Avec Mobile IPv6, un nœud mobile doit effectuer une détection de duplication d'adresse avant d'informer son agent mère de sa nouvelle adresse temporaire. Cette étape est nécessaire car le version non modifiée de Neighbor Discovery est utilisée. L'adresse obtenue par le nœud avec Proxy Mobile IPv6 est toujours la même quelque soit son réseau. Cette phase de détection de duplication n'a donc plus lieu d'être. De plus, au sein de réseaux 6LoWPAN, le protocole Neighbor Discovery a été modifié pour s'adapter aux spécificités de ces réseaux. En particulier, la phase de détection de duplication d'adresse a été supprimée (uniquement si l'EUI-64 est utilisé pour construire l'adresse IPv6 temporaire). En utilisant la version modifiée du protocole, il est fort possible que la durée du handover de Mobile IPv6 soit bien plus courte, comme nous le montrerons dans le chapitre 5.

Dans [73], les auteurs proposent une combinaison des qualités de Fast Handovers for Mobile IPv6 et Proxy Mobile IPv6 nommée Inter-MARIO. L'objectif d'Inter-MARIO est de réduire

le temps de handover des nœuds mobiles en le préconfigurant. Pour cela, ils ont introduit une nouvelle entité : le nœud partenaire, qui va servir de point d'accès aux nœud mobiles. Un nœud partenaire est capable de détecter la mobilité d'un nœud en utilisant des mécanismes tels que la qualité du signal radio (RSSI ou LQI) ou un échange régulier de messages. Ces nœuds disposent également une connaissance étendue de leur voisinage. Lorsqu'un nœud mobile arrive hors de portée du nœud partenaire auquel il est attaché, ce dernier va transmettre des informations à propos des réseaux environnants (fréquence radio de IEEE 802.15.4, par exemple).

Dans un second temps, il va prévenir le 6LBR du prochain réseau du nœud mobile de son arrivée. Une fois que le nœud mobile est associé au nouveau réseau, son routeur va informer le 6LBR de sa présence. Si le 6LBR dispose d'informations de préconfiguration, il effectue la procédure d'association de Mobile IPv6 à la place du nœud mobile. Lorsqu'il réceptionne un message *binding update* envoyé par le nœuds mobile, il intercepte ce message et envoie directement la réponse au nœud mobile. Dans le cas contraire, l'échange entre l'agent mère et le nœud mobile reste inchangé par rapport à Mobile IPv6. Ces différents échanges sont résumés dans la figure 20. La durée du handover de ce protocole a été évaluée par simulation. Leur simulation n'utilise pas de mécanisme de détection de mouvement, celle-ci est effectuée manuellement. La durée du handover est courte mais l'évaluation ne distingue pas les durées des différentes phases du protocole. Il est donc impossible de déterminer l'impact du mécanisme de détection du mouvement d'Inter-MARIO. De plus, l'activation du handover est effectuée manuellement, ce qui influe fortement sur les paramètres évalués.

L'un des désavantages de Proxy Mobile IPv6 est la limitation des nœuds mobiles au sein du domaine Proxy Mobile IPv6 auquel ils appartiennent. Les équipements composant l'Internet des Objets utilisent des applications aussi diverses que variées et cette limitation peut donc être problématique. Cependant, dans certains déploiements les nœuds mobiles disposent d'une mobilité limitée. La restriction de Proxy Mobile IPv6 peut ne pas poser de problème. Dans [74], les auteurs étudient le cas d'un réseau de capteurs déployé au sein d'un hôpital dans lequel le déplacement des nœuds mobiles est limité aux différentes chambres. L'algorithme de support de la mobilité pro-

posé est une version modifiée de Proxy Mobile IPv6 et s'affranchit du protocole Neighbor Discovery. Le nœud mobile initialise la procédure de handover en transmettant une requête d'association 802.15.4. Avant d'y répondre, le 6LBR effectue l'association avec l'équipement gérant le domaine. Une fois l'association terminée, le 6LBR répond à l'association 802.15.4 en incluant l'identifiant sur 16 bits attribué au nœud mobile. Le nœud peut ensuite construire son adresse IPv6 globale qui sera toujours identique. Cette proposition a été évaluée mathématiquement et se dit meilleure que Mobile IPv6 car moins de messages sont transmis. Mais cette étude n'évalue pas la durée du handover ni l'impact du mécanisme de détection de mouvement.

3.4 CONCLUSION

Afin de maintenir ses communications lorsqu'il se déplace, un nœud mobile doit utiliser un protocole de support de la mobilité. Dans le cas contraire, tout paquet émis vers l'ancienne adresse du nœud mobile sera perdu car elle est devenue obsolète. De plus, lorsque le nœud mobile souhaitera utiliser sa nouvelle adresse, ses correspondants réinitialiseront leur session car ils penseront que le paquet provient d'un autre nœud.

Afin d'éviter cette réinitialisation, le protocole Mobile IPv6 a été proposé par l'IETF. Il permet de maintenir la connectivité durant le déplacement du nœud mobile en utilisant un équipement intermédiaire. Différentes extensions ont été proposées pour améliorer les performances de Mobile IPv6, notamment l'utilisation de nouveaux équipements dans les réseaux visités. L'avantage principal de Mobile IPv6 est l'absence d'infrastructure à déployer dans chaque réseau. Il requiert uniquement un agent mère situé dans le réseau mère du nœud mobile. Mais sa consommation énergétique peut augmenter puisque le nœud mobile participe activement à la gestion de sa mobilité.

Différentes évaluations ont été effectuées sur l'utilisation de ces protocoles de support de la mobilité au sein des réseaux de capteurs. Ces évaluations, pas toujours effectuées sur des équipements réels, ne tiennent pas compte des contraintes du processeur, mémoire et énergie. De plus, elles manquent de détails et ne tirent pas parti des derniers standards définis par le groupe de travail 6LoWPAN. D'autre part, ces évaluations

reprochent à Mobile IPv6 la lenteur de son handover. Nous ne pouvons pas statuer sur cette lenteur, sans une expérimentation complète sur capteurs en utilisant les derniers standards.

Pour maintenir sa connectivité alors qu'il se déplace de réseau en réseau, un nœud mobile doit s'équiper de plusieurs protocoles. Au niveau 2, l'utilisation d'un standard est recommandée car ce dernier est susceptible d'être utilisé dans un plus grand nombre de réseaux. Pour être capable de communiquer à l'extérieur d'un réseau, une connectivité de niveau 3 est requise. Pour cela, la couche d'adaptation de 6LoWPAN, adaptant IPv6 aux contraintes des réseaux de capteurs sans fils, est utilisée. Cette couche permet d'avoir une connexion de bout en bout sans requérir de mécanisme de traduction. Cependant, pour maintenir cette connectivité de niveau 3 durant les déplacements, l'utilisation d'un protocole de support de la mobilité, tel que Mobile IPv6 est nécessaire.

Dans le chapitre 4, nous allons présenter le protocole Mobinet utilisant la sur-écoute liée au médium radio pour supporter la mobilité au niveau 2. Le manque d'informations des évaluations présentées dans ce chapitre nous a fait remettre en question leur résultats et donc nous avons décidé de faire notre propre expérimentation de Mobile IPv6 sur une plate-forme de capteurs sans fil. Le design et l'analyse des résultats de ces expérimentations est présenté dans le chapitre 5. Cette analyse met en avant que Mobile IPv6 est parfaitement utilisable sur des réseaux de capteurs sans fil mais nécessite la mise en place d'un nouveau mécanisme de détection de mouvement. Nous présenterons dans le chapitre 6 un nouveau mécanisme de détection de mouvement se basant sur le protocole Mobinet et nous l'évaluerons sur la plate-forme de capteurs sans fil.

Deuxième partie

ÉCOUTER POUR MIEUX SE DÉPLACER

4

GESTION DE LA MOBILITÉ AU NIVEAU 2

Sommaire

4.1 INTRODUCTION

Dans un premier temps, nous nous sommes attachés à améliorer la gestion de la mobilité des nœuds mobiles au sein des réseaux 6LoWPAN. Dans ce chapitre, nous nous focaliserons sur la gestion de la mobilité au niveau 2. Lorsqu'un nœud mobile entre dans un nouveau réseau 6LoWPAN, il doit être en mesure de communiquer au sein de ce réseau. Le protocole IPv6 et les mécanismes sous-jacents assurent que le nœud pourra communiquer au niveau IP. Au niveau 2, quelque soit le type de mécanisme de routage utilisé (*mesh-under* ou *router-over*), le nœud mobile a besoin de connaître son prochain saut pour transmettre ses données. De manière générale, c'est le protocole de routage qui est chargé de fournir cette information. Hors si un nœud mobile se déplace dans un nouveau réseau dont il n'a aucune information, il est peu probable que ce dernier dispose

du même protocole de routage que le réseau, compte tenu du grand nombre de protocoles de routage existants [26].

Ce chapitre s'articule autour de la définition du protocole Mobinet [75, 76]. L'objectif principal de ce protocole est la détection du voisinage d'un nœud mobile en utilisant la sur-écoute liée au médium radio. Dans un premier temps, nous allons détailler le problème et l'impact des deux types de protocoles de routage. Nous présenterons ensuite le fonctionnement du protocole Mobinet et nous conclurons par son évaluation.

4.2 DÉFINITION DU PROBLÈME

Hypothèse : assumons que les nœuds mobiles et les réseaux visités soient capables de communiquer au niveau MAC. Même s'ils n'utilisent pas le même protocole de routage, les nœuds mobiles devraient être capables de transmettre des données au réseau visité. Cependant en raison de leur mobilité, des pertes de paquets sont possibles [77]. En effet durant la transmission d'un paquet, le nœud mobile peut se déplacer et ainsi ne plus être à portée radio de son prochain saut.

Si le réseau 6LoWPAN utilise un mécanisme de routage de type *mesh-under*, on a donc un modèle de routage de type *convergecast* : tous les paquets de données sont transmis vers le 6LBR. Le protocole est uniquement chargé de définir le prochain saut en direction du 6LBR. Dans un tel scénario, lorsqu'un nœud réceptionne un message, il le retransmet à son prochain saut et ainsi de suite jusqu'à ce que le message atteigne le 6LBR. Avec cette hypothèse, il serait donc possible pour des nœuds mobiles de simplement diffuser les données vers les nœuds fixes. Tous les nœuds fixes, se trouvant à portée radio du nœud mobile, vont simplement transmettre les paquets vers le 6LBR (voir figure 23). L'utilisation de nœuds mobiles dans le processus de routage peut poser un problème puisse qu'ils peuvent partitionner le réseau si ces derniers se déplacent. Le protocole de routage devra être plus dynamique pour compenser le partitionnement et consommera donc plus d'énergie. Le RFC 5867 [78] préconise de ne pas intégrer les nœuds mobiles dans le processus de routage. En conséquence, un nœud mobile n'a pas besoin d'implémenter le protocole de routage utilisé dans le réseau visité.

Cette solution, référencée par la suite comme *méthode par diffusion*, consomme peu de ressources du nœud mobile. Dès qu'une transmission est requise, le nœud mobile doit uniquement allumer sa radio et diffuser son message avant d'éteindre à nouveau sa radio. En conséquence, le nœud mobile ne prendra jamais part au routage des données des autres nœuds. Cette solution peut sembler idéale du point de vue des nœuds mobiles mais il y a deux inconvénients majeurs. Transmis par diffusion, les paquets ne sont jamais acquittés et donc le nœud mobile ne peut pas avoir la confirmation de leur réception par le réseau fixe. De plus, l'utilisation de la méthode par diffusion peut sérieusement dégrader les performances du réseau visité. Les communications par diffusion peuvent entrainer la duplication des paquets et, en conséquence, augmenter le trafic au sein du réseau. Ce trafic supplémentaire peut causer la perte de paquets, augmenter la compétition entre les nœuds pour accéder au réseau et augmenter la consommation énergétique du réseau visité.

Cette solution reste applicable avec un mécanisme de routage *route-over* dans le cadre de la transmission de paquets de données. Tout nœud recevant une trame émise par diffusion va la retransmettre vers son prochain saut IP. Mais contrairement à un mécanisme *mesh-under*, un nœud mobile peut s'enregistrer auprès d'un 6LR. Lorsqu'il souhaitera enregistrer à nouveau son adresse IPv6 ou mettre à jour les différentes durées de vie de Neighbor Discovery, il va transmettre ses messages de signalisation vers le 6LR. Ces messages, étant limités à un saut IP, peuvent ne plus être réceptionnés par le 6LR et le nœud mobile devra recommencer toute la procédure d'auto-configuration et d'enregistrement.

4.3 LE PROTOCOLE MOBINET

Le protocole Mobinet consiste à identifier de manière passive le voisinage d'un nœud mobile en utilisant de la sur-écoute. L'objectif est de déterminer le meilleur voisin (en accord avec les métriques du protocole de routage du réseau visité) pour transmettre des données. Le protocole de routage peut utiliser plusieurs critères pour transmettre des données jusqu'au 6LBR : le prochain saut le plus proche géographiquement, celui le plus proche en nombre de sauts [26] ou encore celui pour lequel

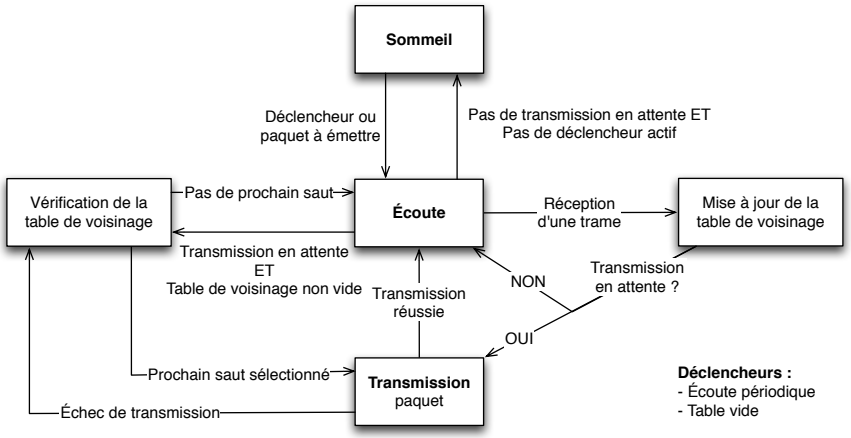


FIGURE 21: Diagramme états-transitions de Mobinet

il reste le plus d'énergie dans sa batterie [79]. Notre solution s'intègre dans un modèle de communication convergecast dans lequel tous les équipements sont capables de communiquer au niveau MAC.

La transmission des données avec Mobinet est effectuée en deux étapes. La première étape consiste à constituer une table de voisinage (section 4.3.1). La seconde étape utilise ensuite cette table pour sélectionner le prochain saut pour la transmission en attente (section 4.3.2).

4.3.1 *Écoute du réseau*

Lorsqu'un nœud mobile entre dans un réseau visité, il va construire une table de de voisinage qui liste les identifiants des nœuds localisés dans son voisinage. Cette table est construite et gardée à jour en écoutant de manière passible les communications environnantes. L'identifiant d'un nœud est acquis généralement en utilisant le champs source de l'en-tête MAC. A chaque entrée de la table est associé un champ Time To Live (TTL). Lorsque le TTL d'une entrée arrive à expiration, cette entrée est retirée de la table. Cela permet aux nœuds mobiles de supprimer les vieilles entrées correspondantes aux voisins qui peuvent être hors de portée.

Le composant le plus consommateur en énergie au sein d'un nœud capteur est sa radio [35]. Utiliser la radio pour écouter les communications environnantes au lieu de limiter son utili-

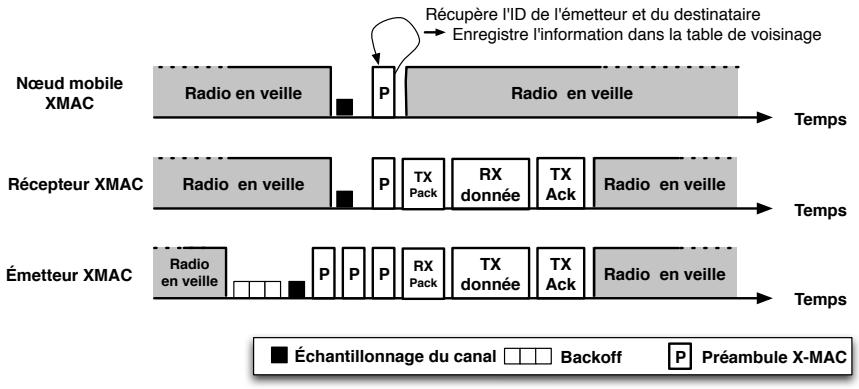


FIGURE 22: Écoute du voisinage avec des protocoles MAC à préambule

sation à la transmission et réception de données le concernant peut rapidement consommer toute l'énergie de la batterie d'un nœud. Pour limiter la consommation énergétique, nous misons sur les longues périodes de sommeil. Nous appelons un processus d'écoute un couple formé par une longue période de sommeil et un déclencheur. Le réveil d'un nœud est déclenché par l'un des processus suivants : *écouter pour transmettre*, *table vide* et *écoute périodique*.

Le premier déclencheur, *écouter pour transmettre*, réveille uniquement la radio lorsqu'un nœud mobile doit transmettre un message et attend d'intercepter une communication. Dès que le nœud mobile a identifié et transmis avec succès le message vers un voisin, il éteint à nouveau sa radio. Le second déclencheur, *table vide*, s'assure que la table de voisinage dispose toujours d'une entrée valide. Il réveille la radio lorsque la dernière entrée de la table expire. La radio est gardée allumée jusqu'à ce qu'une communication soit interceptée. Le dernier déclencheur, *écoute périodique*, active la radio à intervalles réguliers pour écouter les communications environnantes durant une durée prédéterminée. S'il n'y a aucune entrée dans la table lorsque le nœud doit transmettre un message, Mobinet bascule sur le premier déclencheur. Lorsque le message est transmis, le processus initial est restauré. Les différents processus sont détaillés dans la figure 21.

Avec une grande majorité de protocoles MAC, le nœud mobile doit réceptionner le message en entier avant de pouvoir en extraire sa source et sa destination. Cette écoute peut durer durant une longue période et en conséquence consommer beau-

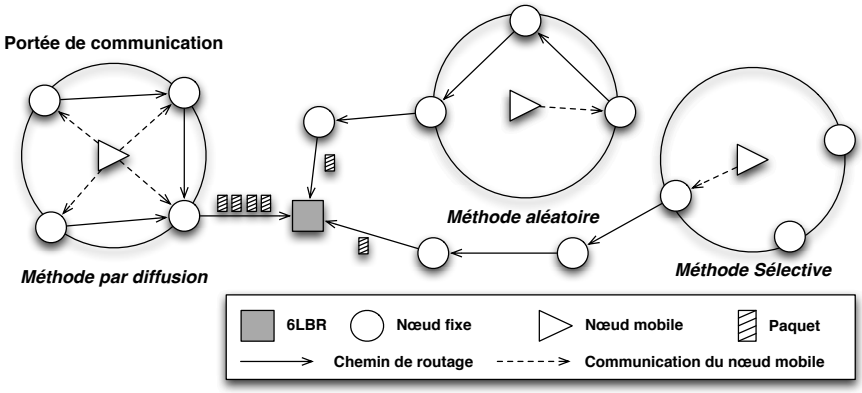


FIGURE 23: Sélection du prochain saut

coup d'énergie. Cette période d'écoute peut cependant être réduite grâce aux protocoles MAC à préambule tels que X-MAC. X-MAC utilise des micro-trames comme préambule contenant les informations nécessaires pour compléter la table de voisinage. La figure 22 illustre l'écoute du voisinage au niveau MAC lors de l'utilisation du protocole X-MAC.

4.3.2 *Sélection du prochain saut*

Dès qu'un nœud mobile veut transmettre un message vers le 6LBR du réseau visité, il va envoyer ce message unicast vers l'un des voisins listés dans la table de voisinage. Ceci est rendu possible par le modèle de routage convergecast et l'absence d'en-tête de routage. Le voisin retransmettra naturellement le message vers le 6LBR en accord avec le protocole de routage utilisé dans le réseau visité. Notre première approche, *méthode aléatoire*, va sélectionner de manière stochastique un des voisins disponibles dans la table de voisinage. Cette approche est présentée dans la figure 23.

Lorsque la table de voisinage contient plusieurs entrées, il est possible pour un nœud mobile d'identifier la direction des communications (à partir de la source et la destination du message) et il peut donc recréer la hiérarchie au sein de sa table de voisinage. Il devrait ainsi être capable de transmettre directement les données au *meilleur nœud* de son voisinage. Le terme meilleur nœud est un terme générique qui varie en fonction du protocole de routage utilisé dans le réseau visité (par exemple, le nœud qui minimise le nombre de sauts vers le 6LBR). Cette

seconde approche est nommée *méthode sélective* et est représentée dans la figure 23.

Ces deux approches de notre solution transmettent des données en établissant une communication unicast. Cela évite la possible duplication de messages qui est inhérente à la méthode par diffusion.

Procédure 1 Algorithme de sélection sélection du prochain saut de Mobinet

```
si methode = aleatoire alors
    prochain_saut ← entreeAleatoire(table).source
sinon
    entree ← entreeAleatoire(table)
    tant que entree.destination ≠ −1 faire
        pour e dans table faire
            si entree.destination == e.source alors
                entree ← e
                stop
            fin si
        fin pour
    fin tant que
    prochain_saut ← entree.source
fin si
retourne prochain_saut
```

4.4 ÉVALUATION

Nous avons implémenté le protocole Mobinet dans le simulateur WSNet [80]. WSNet est un simulateur à événement discrets dédiés à l'étude de réseaux de capteurs sans fil. Nous avons comparé les performances de Mobinet (avec ses 2 méthodes de sélection) avec celles de la méthode par diffusion qui est présenté dans la section 4.2. La simulation est également utilisée pour analyser les différents processus d'écoute définis pour Mobinet et leur impact sur la sélection du prochain saut.

4.4.1 *Environnement de simulation*

Notre scénario de simulation implique 50 nœuds mobiles se déplaçant dans un réseau visité qui est déployé dans une zone

de 100 m x 100 m. Le réseau visité est constitué d'une grille de nœuds de 10 × 10. Chaque nœud est placé à neuf mètres de ses voisins avec une portée de 15 m. Le 6LBR est placé dans un coin de la grille. X-MAC, un protocole MAC basé sur l'échantillonnage de préambules, est implémenté sur chaque nœud (fixe comme mobile). Comme expliqué dans la section 4.3.1, X-MAC inclut la source et la destination du message dans ces morceaux de préambule. Cela devrait réduire la compétition entre les nœuds pour accéder au médium et permettre à notre proposition de réduire sa consommation en n'écoutant qu'une partie du préambule. Le réseau visité utilise un protocole de routage par gradient [81] pour sélectionner le prochain saut en direction du 6LBR. Les nœuds du réseau visité ont une application orientée événements. Ils ne transmettent aucune information tant qu'un événement n'a pas été capturé. Au début de la simulation, nous avons distribué 3600 événements durant les 2 heures de temps simulé (divisés en slots d'une seconde) en utilisant un processus de Poisson :

$$\lambda_{\text{poisson}} = \frac{3600}{7200}$$

Cela correspond donc à 3600 événements distribués sur 7200 slots. La localisation des événements est distribuée uniformément dans la grille de capteurs. Lorsqu'un événement est capturé, un nœud commence à transmettre des paquets de données de 4 octets vers le 6LBR toutes les secondes durant la durée de l'événement. Un événement dure 10 secondes.

Chaque nœud mobile implémente une application basée sur le temps qui transmet 4 octets de données toutes les 4 minutes et se déplace en utilisant un modèle de mobilité à direction aléatoire modifié [82]. Dans ce modèle modifié, un nœud mobile se déplace telle une boule de billard. Lorsqu'il atteint un bord de la zone, il rebondit en utilisant le même angle d'incidence et garde la même vitesse. La vitesse d'un nœud mobile est choisie uniformément entre 0 et 3 m/s. Une entrée de la table de voisinage de Mobinet a un TTL de 10 secondes. Cette durée représente le temps minimum pour un nœud mobile pour traverser la zone de transmission d'un nœud fixe. La table 4 reprend tous ces paramètres utilisés dans la simulation.

Le modèle énergétique a été choisi pour émuler la consommation énergétique d'un composant CC1101 [37] (les différentes

Paramètres simulés	Valeurs
Topologie	Grille carrée (100 m x 100 m) de 100 (10x10) nœuds fixes
Période d'envoi des données	Nœuds fixes : lors d'un événement, chaque seconde durant 10 s Nœuds mobiles : toutes les 4 minutes
Modèle de routage	Routage par gradient
Modèle MAC	X-MAC avec un préambule de 100 ms avec des acquittements
Modèle radio	Fréquence : 868 Mhz ; Portée : 15 m
Modèle énergétique avec une batterie de 3V	Idle : 1.6 mA, RX : 14.6 mA, TX : 16.4 mA, Radio init : 8.2 mA
Nombre d'événements	3600
Durée et nombre de simulations	2 heures, simulées 100 fois pour chaque combinaison
Mobilité	modèle à déplacement aléatoire modifié Vitesse maximale : 3 m/s
Mobinet	TTL d'un voisin : 10 s

TABLE 4: Paramètres de simulations de Mobinet

valeurs de consommation énergétique sont reportées dans la table 4). Les résultats de consommation énergétique présentés dans la section suivante assument qu'une batterie de 3V est utilisée comme pour les nœuds TelosB [83].

Notre proposition utilise alternativement les différents processus d'écoute définis dans la section 4.3.1. Les durées suivantes sont utilisées pour le processus d'écoute périodique : 1 s, 10 s et 60 s pour les périodes de sommeil et 20 ms et 100 ms pour les périodes d'écoute. Durant les processus d'écoute, X-MAC n'est pas autorisé à éteindre la radio pour capturer toutes les communications potentielles. En outre, si un nœud mobile est en cours de réception d'un message alors que la période d'écoute est terminée, l'arrêt de la radio est retardé jusqu'à la réception complète du message. Lorsqu'un nœud mobile utilise la méthode de sélection aléatoire pour déterminer son prochain saut, la radio est coupée dès la détection d'un voisin. À contrario, elle reste allumée durant toute la période d'écoute avec la méthode sélective. Mobinet est également simulé sans proces-

Identifiant	Processus d'écoute	Paramètres	
0	Méthode par diffusion		
1	Pas de processus d'écoute		
		Sommeil	Écoute
2	Écoute périodique	1 s	20 ms
3	Écoute périodique	1 s	100 ms
4	Écoute périodique	10 s	20 ms
5	Écoute périodique	10 s	100 ms
6	Écoute périodique	60 s	20 ms
7	Écoute périodique	60 s	100 ms
8	Table vide	TTL : 10 s	
9	Écouter pour transmettre		

TABLE 5: Identifiant de simulation et processus d'écoute utilisé dans les simulations

sus d'écoute, pour servir comme référentiel pour l'évaluation de l'efficacité des processus d'écoute. Durant ces simulations, la radio alterne des périodes d'activité et de sommeil comme défini par le protocole MAC. Toutes ces valeurs sont reportées dans la table 5. Cette table contient également des identifiants de simulations (Sim ID) utilisés dans les figures de la section suivante.

4.5 RÉSULTATS DE SIMULATION ET ANALYSE

Chaque processus d'écoute (9 processus, voir table 5) avec chaque mécanisme de sélection du prochain saut a été simulé 100 fois, ce qui correspond à un total de 1800 simulations. Chaque jeu de simulation utilise la même graine pour évaluer les processus dans un environnement similaire. Les résultats présentés dans cette section sont une moyenne de toutes les données collectées durant les simulations. Elles incluent également une période de démarrage de 120 secondes et une période de refroidissement de 60 secondes. L'intervalle de confiance à 95% indique l'exactitude de nos mesures. La correspondance entre le Sim ID (c.f. Section 4.4.1) utilisé dans les figures et les processus d'écoute est reportée dans la table 5.

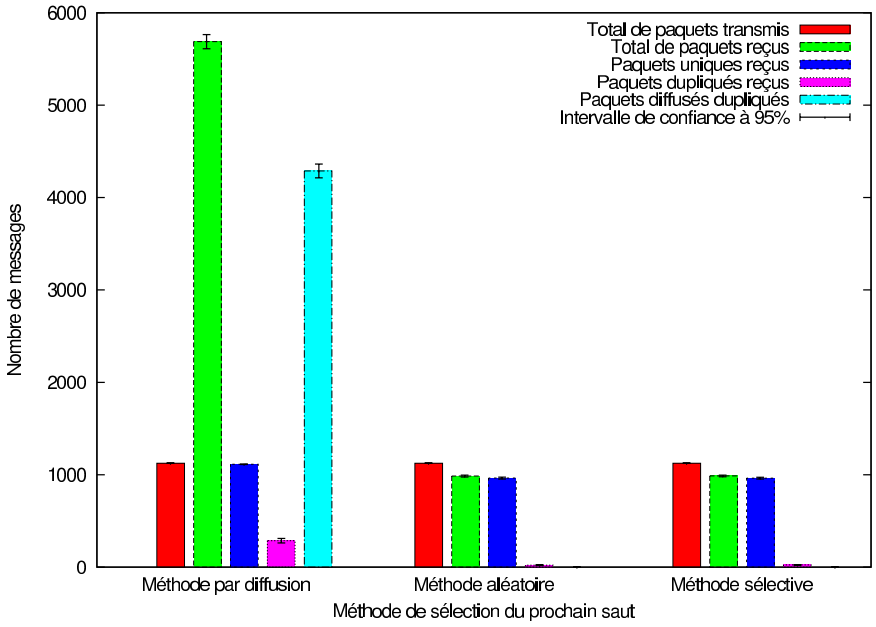


FIGURE 24: Transmission des messages au réseau visité

4.5.1 *Transmission des messages au réseau visité*

1125 messages sont transmis dans chaque simulation par les nœuds mobiles vers le réseau visité, quelque soit le processus d'écoute utilisé. Ces processus n'ayant aucun impact sur le réseau visité, nous avons uniquement analysé l'impact des méthodes de sélection du prochain saut sur le réseau visité. Ceci est rendu possible par l'acquittement des communications unicast. Mobinet est configuré pour transmettre un message jusqu'à ce que ce dernier soit réceptionné par un nœud fixe.

La figure 24 représente les performances de Mobinet et de la méthode par diffusion pour transmettre des messages. Comme prévu, la méthode par diffusion a généré de nombres messages dupliqués (4319 messages reçus sont produits par la diffusion initiale) mais la majorité des messages (1113 messages) sont réceptionnés par le 6LBR. Avec la méthode aléatoire de Mobinet, le 6LBR a réceptionné approximativement 993 messages incluant une moyenne de 967 messages uniques. Des résultats identiques sont observés avec la méthode sélective. Ceci est principalement du à la topologie et au protocole de routage utilisé par le réseau visité. Un prochain saut sélectionné avec la méthode sélective est, au plus, à deux sauts de celui choisi par la méthode aléatoire. Les messages unicast utilisés par Mobinet

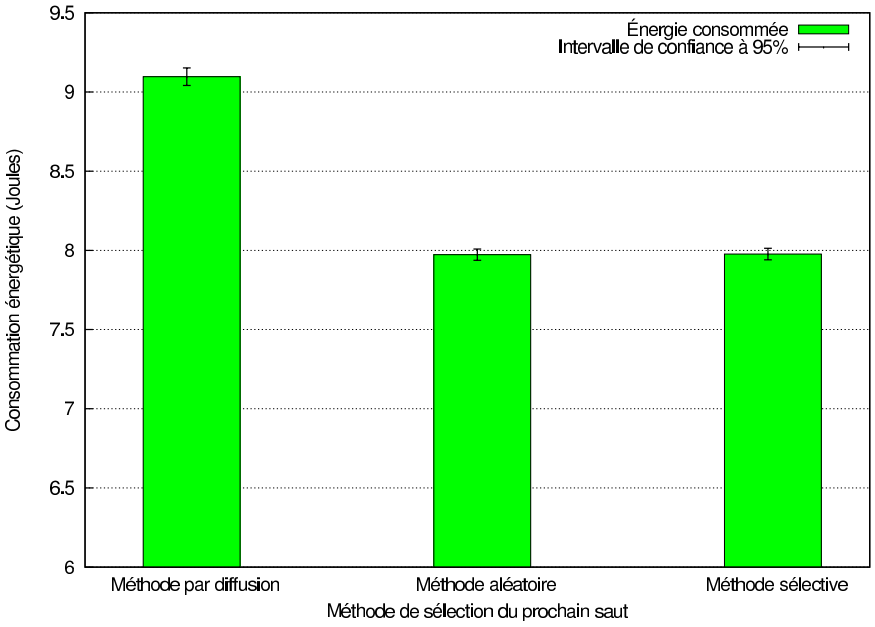


FIGURE 25: Consommation énergétique du réseau visité

peuvent néanmoins générer de la duplication si le nœud mobile ne réceptionne pas l'acquittement du message (en raison de la mobilité par exemple).

Des résultats identiques peuvent être observés sur la consommation énergétique des nœuds fixes du réseau visité. Ces résultats sont présentés dans la figure 25. Comme le même nombre de paquets est transmis au réseau fixe par les nœuds mobiles, le réseau visité va donc consommer la même quantité d'énergie quelque soit le processus d'écoute utilisé par Mobinet.

4.5.2 *Impact de Mobinet sur la consommation énergétique*

La figure 26 représente la consommation énergétique moyenne des nœuds mobiles à la fin de la simulation pour chaque processus d'écoute. Mobinet sans aucun processus d'écoute (Sim ID 1) est utilisé comme référentiel : si la consommation énergétique d'un processus est plus faible que celle expérimentée dans la Sim ID 1, ce processus d'écoute peut être considéré comme efficace. Avec cette version basique de Mobinet, les nœuds mobiles ont consommé respectivement $4,50$ et $4,52$ joules pour les méthodes de sélection aléatoire et sélective. Comme attendu, la méthode par diffusion a consommé le moins d'énergie, seulement $0,06$ joules. L'énergie requise pour transmettre un mes-

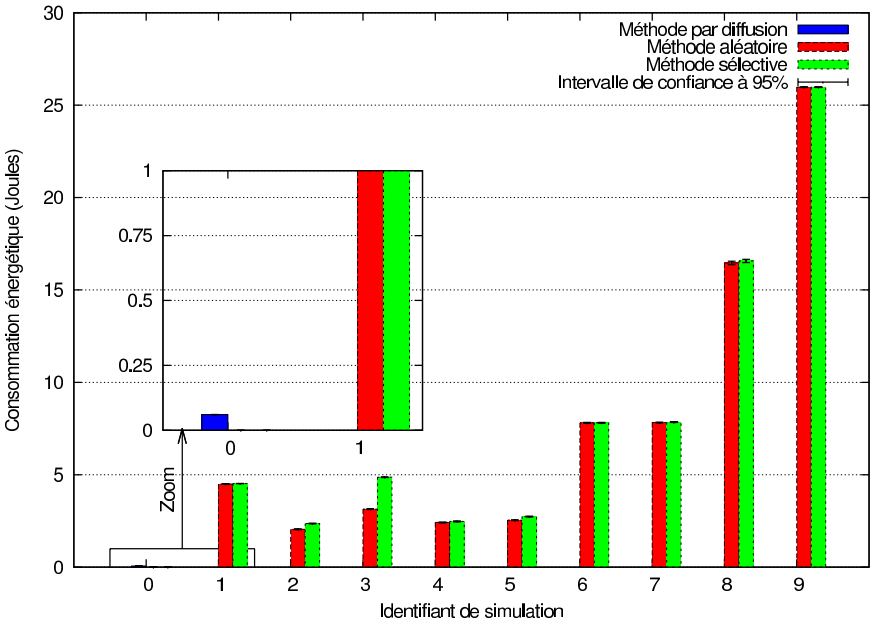


FIGURE 26: Consommation énergétique moyenne d'un nœud mobile

sage dépend principalement du protocole MAC. Avec X-MAC, diffuser des messages implique de transmettre tous les morceaux de préambule avant d'envoyer les données. A l'opposé, les communications unicast permettent au destinataire d'acquitter l'un des morceaux de préambule et ainsi réduire le nombre de morceaux à émettre. Cela réduit la consommation énergétique des nœuds fixes et mobiles. Alors que les processus d'écoute n'ont aucun impact sur la transmission de messages et la consommation énergétique des nœuds fixes, ils sont inégaux face à la consommation énergétique des nœuds mobiles. Pour quatre d'entre eux (Sim ID 6 à 9), la consommation est plus de deux fois supérieure à la version de base de Mobinet. Le processus d'écoute qui a consommé le moins d'énergie est celui de la Sim ID 2 (consommation énergétique moyenne de 2.05 joules) avec une méthode de sélection aléatoire du prochain saut.

Nous pouvons mettre en avant que seule l'écoute périodique avec de petite durée a obtenu une faible consommation énergétique. Lorsque les nœuds mobiles ont utilisé la méthode sélective, ils ont consommé plus d'énergie qu'avec la méthode aléatoire. C'est particulièrement visible sur les Sim ID 2 et 3.

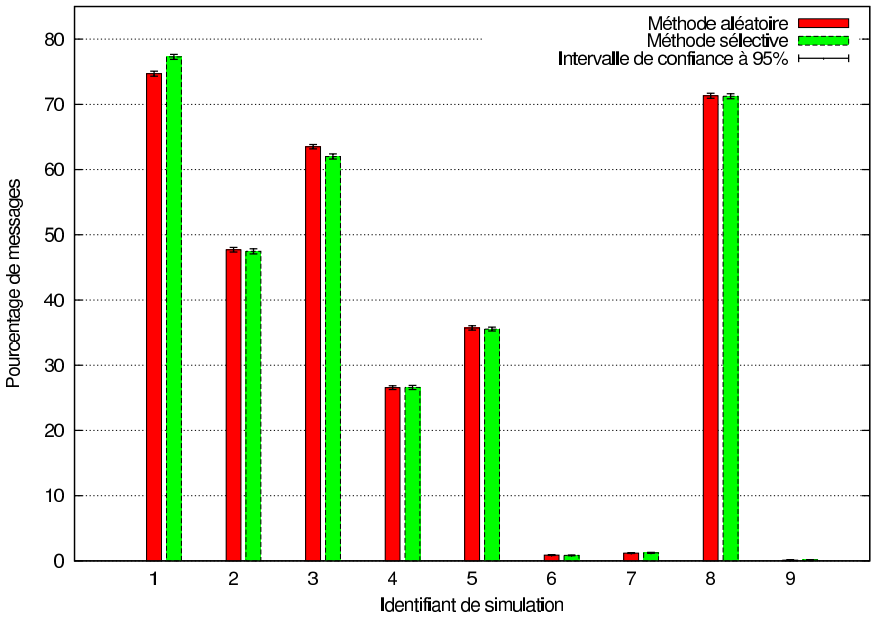


FIGURE 27: Efficacité de l'écoute du voisinage

4.5.3 *Performance des processus d'écoute*

La figure 27 représente le pourcentage de messages envoyés par les nœuds mobiles pour lesquels Mobinet avait au moins une entrée dans sa table de voisinage. La méthode par diffusion (Sim ID 0) n'est pas représentée car elle n'utilise pas de mécanisme d'écoute. Nous avons remarqué que seuls les processus d'écoute périodiques avec de courtes périodes de sommeil sont efficaces énergiquement. Les Sim ID 6 et 7 qui ont de longues périodes de sommeil doivent passer au processus forçant l'écoute du canal (processus écouter pour transmettre) pour la plupart des messages. Ce processus consomme beaucoup d'énergie car la radio est reste allumée durant de longues périodes. La même observation peut être effectuée pour la Sim ID 8 : bien que la table dispose toujours d'une entrée pour chaque message, ce processus a consommé plus d'énergie que les Sim ID 4 et 5 ayant la même période de sommeil. Comme expliqué dans la section 4.3.1, le processus table vide garde sa radio allumée jusqu'à ce qu'un voisin soit détecté tandis que les deux autres processus écoutent uniquement durant 20 ms et 100 ms. Cela confirme que les processus *table vide* et *écouter pour transmettre* sont inefficaces et doivent être combinés avec une écoute périodique pour avoir un effet bénéfique sur la consommation énergétique des capteurs.

Pour conclure, les deux méthodes de sélection du prochain saut de Mobinet sont toutes deux efficaces pour transmettre les messages au réseau visité. Les résultats mettent également en évidence qu'une écoute périodique constituée d'une période de sommeil d'une seconde et une période d'écoute de 20 ms permet de réduire la consommation énergétique d'un nœud mobile.

4.6 CONCLUSION

Dans ce chapitre, nous avons présenté une nouvelle approche appelée Mobinet. Mobinet est un protocole qui gère la mobilité au niveau 2 et permet à un nœud mobile de sélectionner un saut vers qui transmettre des données à l'aide de la sur-écoute. Pour cela, deux mécanismes de sélection du prochain saut sont définis et plusieurs processus d'écoute sont proposés pour réduire la consommation énergétique des nœuds mobiles.

Nous avons évalué son efficacité par simulation. Dans ce scénario, Mobinet doit permettre aux nœuds mobiles de pouvoir transmettre des informations dans un réseau visité inconnu avec la couche MAC comme seul point commun. Les résultats ont montré que Mobinet est efficace pour détecter le voisinage d'un nœud mobile et lui permet de transmettre des données tout en limitant sa consommation énergétique. Le fait qu'il soit basé sur des mécanismes déjà existants est un des avantages de Mobinet.

Un nœud mobile est capable à présent de gérer sa mobilité au niveau 2 grâce à Mobinet. Cependant si le nœud mobile souhaite maintenir sa connectivité avec des correspondants extérieurs alors qu'il se déplace de réseau visité en réseau visité, il est nécessaire de gérer la mobilité au niveau 3. Dans le prochain chapitre, nous allons étudier cette mobilité de niveau 3 en commençant par évaluer les capacités de Mobile IPv6 au sein des réseaux de capteurs sans fil.

5

EVALUATION DE PERFORMANCES DE MOBILE IPv6

Sommaire

5.1 INTRODUCTION

Dans le chapitre précédent, nous nous sommes focalisés sur la mobilité au niveau 2 dans les réseaux visités. Cependant, pour joindre et être joignable par des nœuds correspondants hors du réseau, les nœuds mobiles doivent utiliser un protocole supportant la mobilité au niveau 3. Notre attention s'est donc portée sur Mobile IPv6, le premier protocole permettant de supporter la mobilité au niveau 3 standardisé par l'IETF. Ce protocole dispose de plusieurs avantages : il s'intègre naturellement dans la pile 6LoWPAN et il ne requiert aucune infrastructure supplémentaire à déployer dans les réseaux visités. Ce dernier avantage va permettre aux nœuds mobiles de conti-

nuer à communiquer quelque soit le réseau dans lequel ils se trouvent, supprimant ainsi toute limite à leur mobilité.

Les évaluations effectuées sur ce protocole dans les réseaux de capteurs sans fil sont peu convaincantes (voir chapitre 3). En effet, ces dernières ne prennent pas en compte les dernières améliorations apportées par le groupe de travail 6LoWPAN sur l'intégration d'IPv6 dans les réseaux à faible débit, faible consommation et faible portée. Pour confirmer ou infirmer l'incompatibilité de Mobile IPv6 avec les réseaux 6LoWPAN, nous avons décidé d'effectuer une évaluation complète de ce standard [84]. Cette évaluation va se focaliser sur les délais résultant de chaque opération de Mobile IPv6 et sur l'impact du handover sur les communications montantes et descendantes.

5.2 MOBILE IPV6 ET LA DÉTECTION DE MOUVEMENTS

5.2.1 *La norme*

Mobile IPv6 dépend des messages *router advertisement* pour détecter le mouvement du nœud mobile, c'est-à-dire l'arrivée du nœud dans un nouveau réseau IPv6. Dans les réseaux IPv6 classiques, ces messages sont envoyés de manière périodique par les routeurs ou leur envoi est sollicité par les nœuds (par un message *router solicitation*) lorsqu'une interface est activée. Par défaut, Neighbor Discovery spécifie que deux messages *router advertisement* non sollicités consécutifs soient espacés d'une période allant de 200 à 600 secondes [64]. Ce délai étant trop grand pour détecter rapidement les changements, Mobile IPv6 suggère modifier ce délai pour le réduire à une période allant de 30 ms à 70 ms [29]. Pour les réseaux de capteurs, cette haute fréquence d'émission va consommer trop d'énergie et augmenter la contention sur le médium sans fil car ces messages sont destinés, par défaut, à tous les nœuds IPv6 du lien.

Mobile IPv6 peut également se servir de la détection de connexion à un lien de niveau 2. Cependant, dans les réseaux de capteurs sans fil, une grande partie des protocoles MAC ne génère pas cet événement, comme les protocoles MAC à préambule ou le mode sans balise de la norme IEEE 802.15.4. Ce mécanisme ne peut donc pas être considéré comme viable au sein des réseaux de capteurs sans fil.

5.2.2 *Notre proposition*

Pour détecter le mouvement de nœuds mobiles, nous proposons de nous fier à d'autres mécanismes de Neighbor Discovery. Plusieurs durées de vies sont définies dans [65] : la durée de vie du routeur par défaut, la durée de vie d'un préfixe ou la durée de vie d'une adresse globale. Le temporisateur ayant la durée de vie maximale la plus courte est celle du routeur par défaut, qui est limitée à une durée de 150 minutes. Avant que la durée de vie n'expire, le nœud mobile doit envoyer un message *router solicitation* en unicast vers son routeur par défaut (6LBR ou 6LR) pour obtenir un nouveau message *router advertisement*. Cependant, aucune information n'est donnée dans la spécification sur le nombre d'émissions et le délai entre deux messages *router solicitation* consécutifs tant que le nœud mobile ne reçoit aucun *router advertisement* en réponse. En conséquence, nous proposons d'utiliser la même procédure de retransmission que celle définie pour les *router solicitations* envoyés en multicast : le nombre de retransmissions est fixé à 3 et chaque retransmission est séparée par 10 secondes.

La grande durée de vie du routeur par défaut peut cependant être problématique. En effet, si la valeur maximale est utilisée et que le nœud mobile quitte le réseau visité juste après avoir enregistré son adresse temporaire, il sera déconnecté durant 150 minutes. De plus, une si longue déconnexion ne peut pas être distinguée d'une panne du nœud mobile. C'est pourquoi, nous suggérons d'utiliser une durée de vie beaucoup plus courte. En revanche, cette durée ne doit pas être trop courte pour éviter aux nœuds de consommer trop d'énergie et d'encombrer le médium radio.

Pour résumer, un handover de Mobile IPv6 utilisant notre adaptation se déroule de la manière suivante au sein d'un réseau 6LoWPAN : lorsqu'un nœud mobile arrive dans un nouveau réseau IPv6, il attend que la durée de vie du routeur par défaut expire. Cette expiration déclenche alors la transmission de messages *router solicitations* envoyés en unicast vers l'ancien routeur du nœud mobile. En raison de son déplacement, le nœud mobile ne peut plus réceptionner un *router advertisement* en réponse car son ancien routeur n'est plus joignable au niveau IP. Après 3 tentatives (séparées par 10 secondes), le nœud mobile considère que son ancien routeur est inaccessible et tente d'en découvrir un nouveau en émettant un message *router soli-*

Procédure 2 Algorithme de détection de la mobilité

Évenement : expiration durée de vie du routeur par défaut
  mobilité ←**vrai**
  pour i de 1 à 3 **faire**
    envoi *router solicitation* en unicast
    initialise temporisateur à 10 secondes
    attend *router advertisement* ou expiration temporisateur
    si réception *router advertisement* **alors**
      mobilité ←**faux**
      stop
    fin si
  fin pour
  si *mobilité* **alors**
    suppression informations précédent réseau
    envoi *router solicitation* en multicast
  fin si

citation en multicast. Après avoir réceptionné un message *router advertisement* d'un nouveau routeur, le nœud mobile construit une adresse IPv6 temporaire à l'aide de l'auto-configuration sans état. Cette nouvelle adresse IPv6 temporaire ne sera valide uniquement dans le nouveau réseau. Si cette adresse est enregistrée avec succès auprès du routeur, le nœud mobile envoie finalement un message *binding update* vers son agent mère et attend son acquittement.

5.3 ÉVALUATION

Nous avons implémenté Mobile IPv6 dans Contiki [85] (version 2.5). Contiki est un système d'exploitation libre et gratuit écrit en langage C. Il est conçu pour être déployé sur des systèmes embarqués et les réseaux de capteurs sans fil. Dans notre étude, l'intérêt de Contiki réside dans sa pile uIPv6 [86] qui est certifiée IPv6 Ready Phase 1. Cela certifie qu'un nœud utilisant la pile uIPv6 inclut les protocoles de base d'IPv6 et est interopérable avec d'autres implémentations IPv6. Cette pile implémente déjà une partie du mécanisme de compression LOWPAN_IPHC et le protocole Neighbor Discovery pour réseaux IPv6. Nous avons ajouté les éléments manquants au mécanisme de compression à savoir la gestion de l'encapsulation IPv6 dans IPv6 ainsi que le champ LOWPAN_NHC qui sont utiles pour

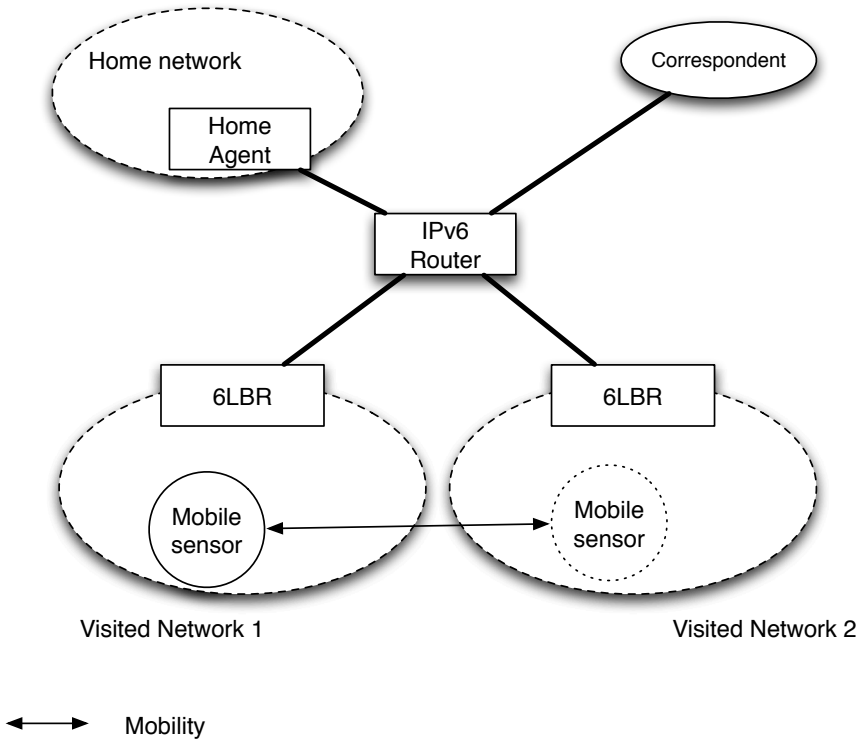


FIGURE 28: Vue schématique de la plate-forme

le tunnel utilisé par Mobile IPv6. Nous avons également modifié Neighbor Discovery pour le rendre conforme à la version adaptée pour les réseaux 6LoWPAN (voir chapitre 2.5). Ceci représente un ajout d'environ 2600 lignes de code.

5.3.1 *Spécifications de la plate-forme*

La plate-forme expérimentale que nous avons déployée pour évaluer notre version de Mobile IPv6 implique deux réseaux visités déployés dans une pièce carrée (12 m × 12 m). Chaque réseau visité est composé d'un 6LBR. Durant une expérience, un nœud mobile se déplace entre ces deux réseaux visités. Une vue d'ensemble de la plate-forme d'expérimentation est illustrée dans la figure 28. Les équipements utilisés par le nœud mobile et les 6LBR sont des TelosB [83] développés par Crossbow. Un nœud TelosB inclut un module émetteur/récepteur compatible avec le standard IEEE 802.15.4 au niveau physique et MAC. 6LoWPAN [27] a été défini pour permettre la transmission de paquets IPv6 sur les réseaux 802.15.4. Afin de réduire l'impact de la couche MAC sur l'évaluation des performances des cou-

Paramètres expérimentaux	**Valeurs**
Nœuds déployés	1 nœud mobile
Organisation de la plateforme	2 réseaux visités avec 1 6LBR et entre 0 et 7 autres nœuds
Espacement des réseaux	10 mètres
Modèle applicatif	*Basé sur le temps* : CBR de 1 message toutes les secondes
Taille des paquets de données	de 20 à 36 octets
Paramètres de 6LoWPAN	**Valeurs**
Durée de vie d'un routeur	30 secondes
Expiration des *router solicitation* en unicast	1 seconde
Paramètres de 802.15.4	**Valeurs**
Fréquence	2.4 GHz
Mode MAC	Mode avec contention
Tentatives maximales	3
Exposant de backoff	de 3 à 5
Taille de la file de messages	10 paquets
Puissance d'émission	-25 dBm (portée : ± 3.5 m)

TABLE 6: Spécifications de l'expérimentation

ches supérieures, nous avons décidé d'utiliser le mode sans balise de la couche MAC définit par le standard IEEE 802.15.4. Dans ce mode, les nœuds peuvent transmettre des données à tout moment et doivent utiliser le mécanisme CSMA/CA pour déterminer si le médium radio est libre ou non.

Chaque nœud implémente la couche d'adaptation d'IPv6 utilisant le mécanisme de compression LOWPAN_IPHC, défini dans 6LoWPAN [29], pour compresser les en-têtes et le protocole Neighbor Discovery adapté pour les réseaux 6LoWPAN. Le réseau visité utilise un protocole de routage *mesh-under* et en conséquence le nœud mobile est toujours à un saut IP du 6LBR même si plusieurs sauts de niveau 2 sont nécessaires pour l'atteindre. Nous avons gardé les valeurs par défaut définies dans les différentes spécifications pour chaque temporisateur.

Le nœud mobile utilise également notre solution pour détecter son mouvement. Le délai entre deux messages *router solicitation* envoyé en unicast consécutifs est fixé à 1 seconde. Nous avons choisi la même valeur que l'intervalle entre deux messages *neighbor solicitation* qui sont utilisés par le mécanisme de détection d'inaccessibilité de voisins. Cette courte période va permettre de réduire le temps de détection du mouvement. De plus, afin de réduire le temps de l'expérimentation, nous avons fixé la durée de vie du routeur par défaut à 30 secondes (la valeur par défaut n'est pas strictement définie, mais peut aller jusqu'à 150 minutes [65]).

Durant le déplacement, il transmet périodiquement des données vers un correspondant distant. Pour créer un trafic applicatif réaliste, nous avons basé notre modèle de trafic sur des mesures réelles effectuées sur des animaux sauvages avec des biologgers. Chaque seconde, le nœud mobile transmet 20 octets de données (correspondant par exemple à des traces ECG ou d'accélération). De plus, 1 message sur 10 inclut également la date et l'heure locale pour une taille additionnelle de 6 octets. Pour finir, toutes les 60 secondes, 10 octets supplémentaires sont transmis incluant des informations globales telles que la température, la luminosité, le niveau de batterie, etc. Toutes ces informations sont résumées dans la table 6. L'architecture logicielle des nœuds capteurs est illustrée dans la figure 29.

5.3.2 *Récupération des statistiques*

Durant nos mesures, nous avons décidé d'enregistrer les événements pertinents se produisant aux différentes couches dans la mémoire du micro-contrôleur, tel que le temps de transmission des différents messages. Nous avons évité d'effectuer une surveillance intrusive qui pourrait perturber le processus normal du nœud. En effet, un traitement lourd dans des interruptions matérielles (par exemple une fonction d'affichage) peut empêcher l'exécution d'une autre interruption (la réception d'un paquet) et peut affecter nos mesures. En conséquence, nous avons décidé de collecter toutes les statistiques des nœuds à la fin de l'expérience et non en temps réel. Dès lors, la prise de mesure n'a pas d'impact sur les temps mis par les différents échanges de messages.

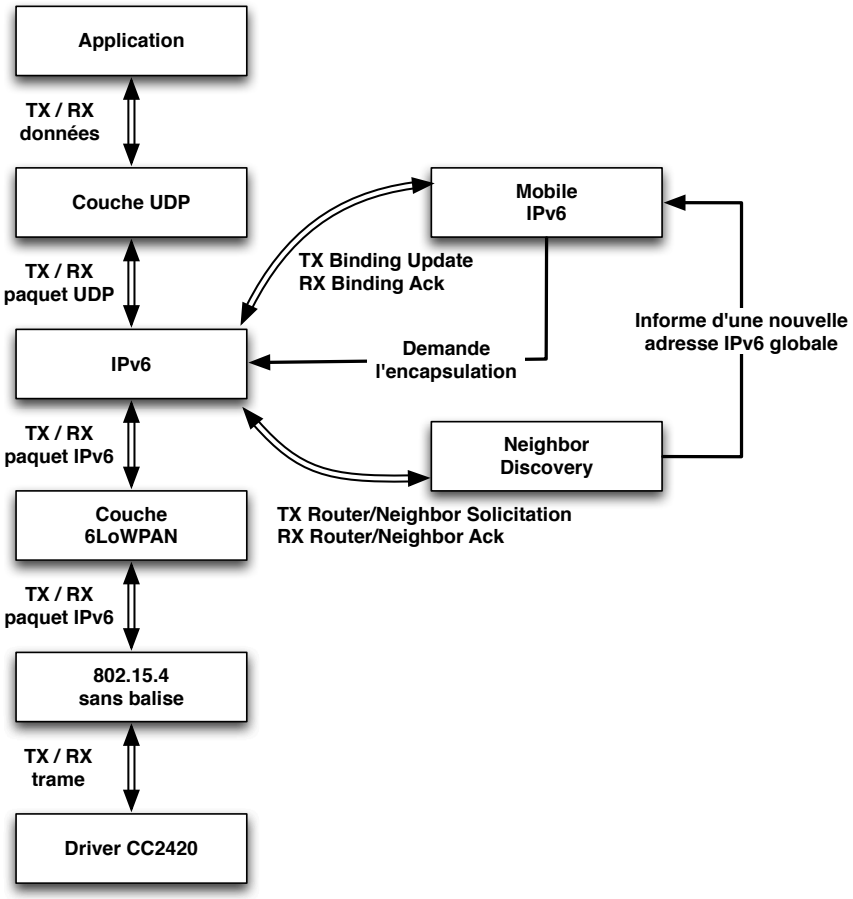


FIGURE 29: Architecture logicielle d'un nœud capteur

Afin de mesurer la consommation énergétique du nœud mobile, nous avons utilisé la librairie de profilage Energest de Contiki [87]. Cette librairie permet de récupérer à un moment donné le temps d'utilisation du processeur lorsqu'il est actif ou endormi ainsi que le temps durant lequel la radio était en mode réception ou en mode transmission. Afin de convertir ces temps en consommation énergétique, nous utilisons le modèle énergétique linéaire présenté dans [87] qui utilise l'équation suivante :

$$Energie = \left(\sum_{i}^{composants} Intensité_i \times temps_i \right) \times Voltage$$

Nous avons utilisé les valeurs d'intensité contenues dans les fiches techniques du composant radio CC2420 [36] et du processeur MSP430 [88]. Le processeur n'étant pas actif à une fréquence constante, nous avons décidé d'utiliser une valeur moyenne comme présenté dans la fiche technique du TelosB [89].

Composant	Mode	Consommation
Processeur	Actif	1.8 mA (min : 400 µA ; max : 3,2 mA)
MSP430	Endormi	5,1 µA
Radio	Réception	19,7 mA
CC2420	Transmission	8,5 mA (à −25 dBm)

TABLE 7: Consommation énergétique des composants d'un TelosB

Les valeurs sont reportées dans la table 7. Il faut également noter que la consommation énergétique va également dépendre du système d'exploitation utilisé [90].

5.4 RÉSULTATS

En plus de l'évaluation de Mobile IPv6, nous avons décidé d'analyser l'impact de la couche d'adaptation d'IPv6 proposée dans 6LoWPAN sur le trafic de signalisation et de données. Nous avons déployé trois micro-logiciels avec différents niveaux de compression. Le premier, *sans compression*, ne compresse aucun en-tête. Il implémente donc les en-têtes complets IPv6 et de Mobile IPv6. Le second micro-logiciel, appelé *sans compression de Mobile IPv6*, compresse uniquement les en-têtes comme défini dans l'encodage LOWPAN_IPHC. Finalement, le micro-logiciel *compression totale* ajoute la compression des en-têtes Mobile IPv6 comme définit dans [68]. Chaque méthode de compression a été jouée 50 fois, pour un total de 150 handovers de niveau 3. L'intervalle de confiance à 95% indique la fiabilité de nos mesures.

5.4.1 *Durée de perte de connexion IP*

Nous avons dans un premier temps évalué la durée de perte de connexion IP. La figure 30 représente le temps moyen, durant lequel la connexion entre le correspondant et le nœud mobile est perdue. Les résultats montrent que le handover de niveau 3 nécessitent environ 19 secondes pour être effectué. Les grands intervalles de confiance sont liés au mécanisme de détection de mouvement. Une fois que le nœud mobile est dans

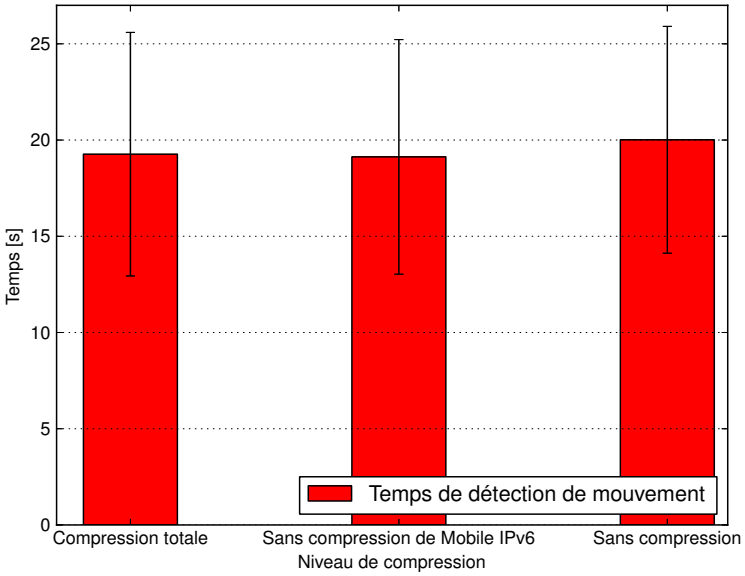


FIGURE 30: Durée de perte de connexion

un nouveau réseau IPv6, il doit attendre l'expiration de la durée de vie du routeur par défaut avant d'être en mesure de détecter son mouvement. Durant nos expériences, cette durée de vie est fixée à 30 secondes (la valeur maximale autorisée par la norme est fixée à 150 minutes). En fonction du moment où le nœud va se déplacer, le délai avant d'envoyer un message *router solicitation* et donc d'initier la détection de mouvement peut varier entre 0 et 30 secondes, ce qui explique les grands intervalles de confiance. Cette première observation montre que le mécanisme de détection de mouvement, comme nous l'avons défini, prend trop de temps pour permettre un handover rapide.

5.4.2 *Temps d'un handover*

La figure 31 détaille le temps requis par le handover de niveau 3 une fois la détection de mouvement terminée. La procédure démarre lorsque le nœud mobile envoie son premier message *router solicitation* en multicast et se termine quand il reçoit un message *binding acknowledgement* positif. Comme nous pouvons l'observer, une compression totale des en-têtes permet de réduire légèrement la latence du handover de 141 ms à 130.6 ms. De plus, nous pouvons remarquer que la compression des

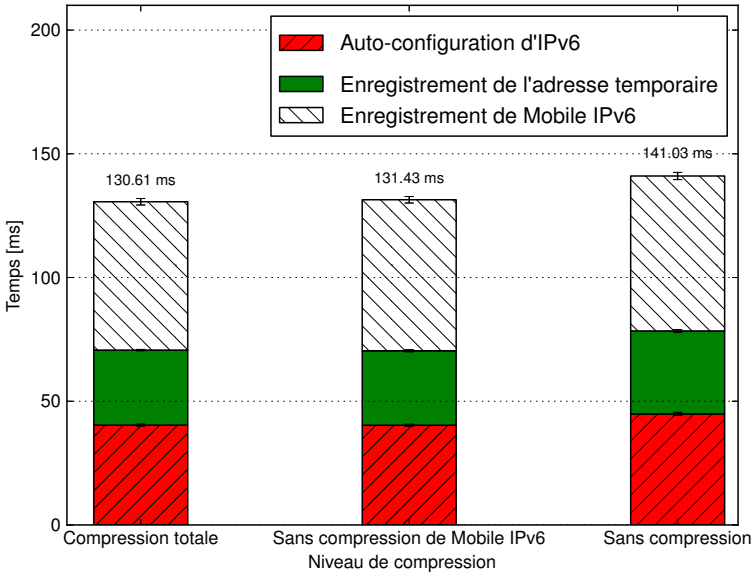


FIGURE 31: Durée des messages de signalisation

en-têtes de mobilité réduit cette latence de seulement 0,6 ms. Ces performances sont principalement dues à l'architecture logicielle du nœud mobile. La pile uIPv6 est construite pour fonctionner sur de multiples plates-formes (allant d'équipements très contraints jusqu'à des plates-formes très puissantes) qui peuvent être directement connectés à des réseaux IPv6. En conséquence, tous les paquets IPv6 sont traités non compressés. Les en-têtes sont ensuite compressés et décompressés par la couche d'adaptation de 6LoWPAN avant la transmission et lors de la réception. Si les paquets étaient traités compressés, la durée de ces échanges pourrait être réduit, en particulier si plusieurs sauts IP sont nécessaires.

La compression / décompression des en-têtes nécessite uniquement l'utilisation du processeur durant une courte période, ce qui retarde légèrement la transmission du paquet. De ce fait, cette phase de compression requiert moins d'énergie que la transmission complète des octets compressés. De manière générale, plus la transmission est longue, plus le risque d'augmenter la contention sur le médium est élevé. Pour conclure, les mécanismes de compression aident à réduire à la fois la consommation énergétique et la contention sur le médium sans fil.

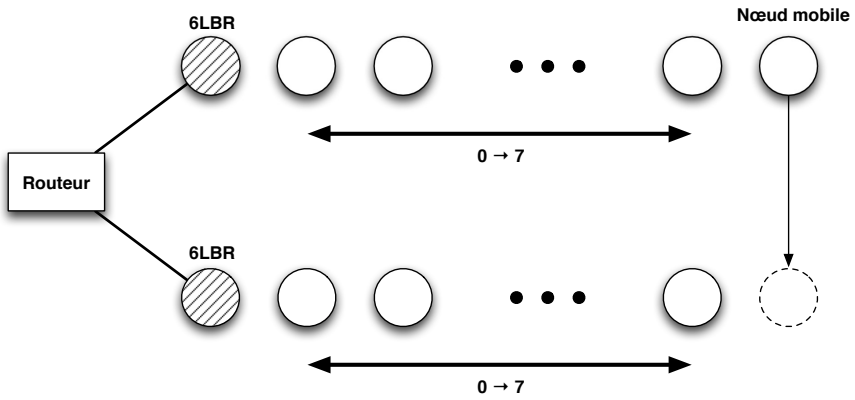


FIGURE 32: Détail de la plate-forme lors de l'évaluation avec plusieurs sauts

5.4.3 *Incidence du nombre de sauts*

Le mécanisme d'auto-configuration sans état d'IPv6 et l'enregistrement de l'adresse IPv6 globale requiert uniquement 70.6 ms pour être effectué avec une compression totale. Ce délai dépend essentiellement du nombre de sauts de niveau 2 entre le nœud mobile et son routeur. Avec un routage *mesh-under*, les échanges liés à Neighbor Discovery sont effectués entre le nœud mobile et le 6LBR qui peut être situé à plusieurs sauts. Ainsi, plus le nœud mobile est éloigné, plus la durée de ces échanges va augmenter. Ceci est d'autant plus important dans les réseaux de capteurs sans fil que dans les réseaux sans fil classiques, car les nœuds sont situés dans la plupart des cas à plus d'un saut du 6LBR. Afin d'évaluer cet impact, nous avons étendu les réseaux visités pour former une ligne de nœuds entre le nœud mobile et le 6LBR pour atteindre un nombre de sauts allant de 1 à 8. La figure 32 schématise les nouveaux réseaux visités. La sélection du prochain saut s'effectue à l'aide d'un routage *mesh-under* statique. La figure 33 représente la durée de chaque phase du handover dans Mobile IPv6 dès lors que la détection de mouvement est terminée. Cette durée croît en fonction du nombre de sauts. Cette croissance plus ou moins linéaire ajoute en moyenne 100 ms par saut pour atteindre un maximum 900 ms lorsque tous les en-têtes sont compressés.

Ces résultats utilisent la couche MAC 802.15.4 en mode à contention. Cette couche MAC offre de bonnes performances car un nœud peut émettre à tout moment en étant certain que ses voisins réceptionneront bien le paquet. Ainsi, si des délais

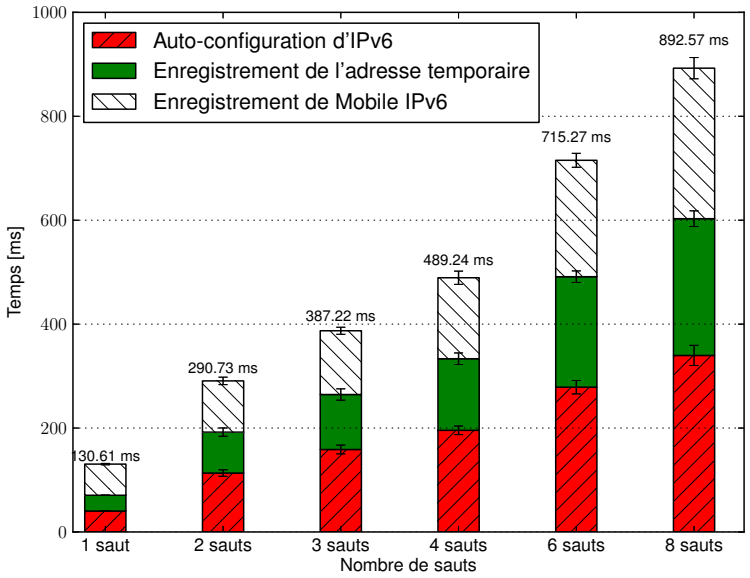


FIGURE 33: Durée des messages de signalisation avec un routage mesh-under

supplémentaires sont induits par la couche MAC, le délai de réponse des messages *neighbor solicitation* (fixé à 1 seconde) peut être dépassé alors que la réponse est en cours de transmission. Le nœud va alors retransmettre le message sans que cela soit nécessaire et par conséquent augmenter le trafic sur le médium radio.

Avec un routage *route-over*, les échanges liés à Neighbor Discovery sont effectués entre le nœud mobile et son routeur qui est situé à un saut que ce soit au niveau 2 et 3. En conséquence, seule la phase de détection de duplication d'adresse (si le nœud mobile utilise une adresse MAC 16 bits pour générer son adresse IPv6 temporaire) et l'enregistrement de Mobile IPv6 sont impactés par la distance entre le 6LR et le 6LBR.

5.4.4 *Impact du handover sur les flux de données*

La figure 34 illustre l'impact du handover de niveau 3 sur les données transmises ou reçues. Chaque point représente la réception ou la transmission d'un paquet au temps indiqué sur l'axe des abscisses. En moyenne, 21 paquets sur 30 sont perdus durant le handover du nœud mobile. Ces pertes interviennent

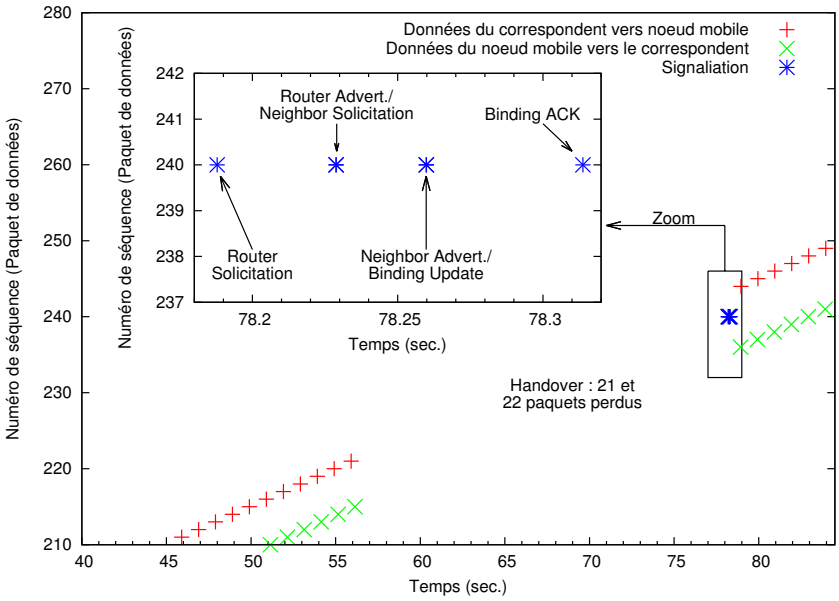
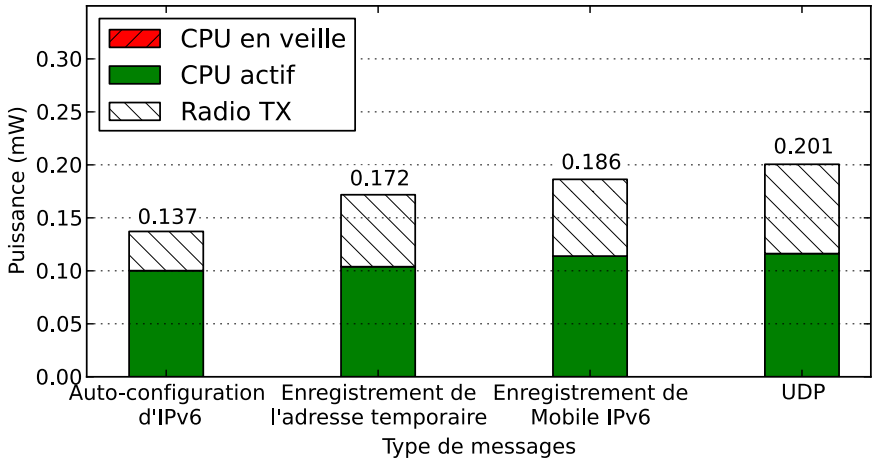


FIGURE 34: Impact du handover de niveau 3 sur les flux de données

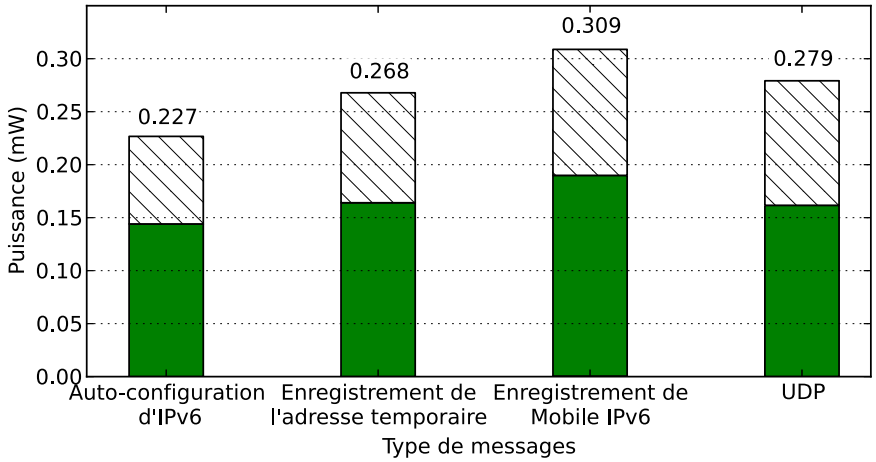
lorsque le nœud mobile quitte le premier réseau visité. Les détails du handover sont illustrés dans la section agrandie de la figure. Comme nous pouvons le constater, les paquets de données sont à nouveau reçus dès que le nœud mobile s'est enregistré auprès de son agent mère. La durée de détection de mouvement du nœud mobile est donc la principale cause de la latence du handover. Réduire la durée de vie du routeur par défaut va permettre d'accélérer cette détection de mouvement car le nœud mobile vérifiera plus souvent s'il peut toujours communiquer avec son routeur. Cependant, ce sera au détriment de la consommation énergétique ou la contention du médium radio dans le réseau 6LoWPAN. Pour conclure, la figure 34 montre également que l'encapsulation / décapsulation a un impact minime sur le temps de transmission des paquets de données.

5.4.5 *Impact sur la consommation énergétique*

Nous avons terminé notre expérimentation par une estimation de la consommation énergétique du nœud mobile. Les figures 5.35(a) et 5.35(b) représentent l'estimation de la consommation énergétique, en utilisant les temps obtenus grâce à la librairie de profilage Energest, pour chaque échange de signalisation et l'énergie requise pour transmettre un message UDP



(a) Compression de tous les en-têtes



(b) Aucune compression d'en-tête

FIGURE 35: Consommation énergétique du nœud mobile

avec ou sans compression des en-têtes. Ces résultats montrent que le mécanisme de compression LOWPAN_IPHC permet de réduire à la fois la consommation énergétique liée à l'utilisation du processeur et à la transmission du paquet. En effet, lors de l'utilisation du micro-logiciel avec une compression totale, la transmission consomme moins car les messages compressés prennent moins de temps à être transmis. Nous pouvons également remarquer que malgré la phase de compression, l'énergie totale consommée par le processeur reste plus faible qu'avec le micro-logiciel *sans compression* car le processeur reste actif durant la phase de transmission.

L'estimation de la consommation énergétique de la radio en mode réception n'est pas représentée. En raison du protocole

MAC utilisé, la radio est toujours en mode réception. Elle a une consommation fixe estimée à 55,8 mW. Les résultats présentés utilisent la pile uIPv6 qui traite des paquets IPv6 complets avant des les compresser pour la transmission. On peut encore espérer de meilleurs résultats avec une pile totalement optimisée pour 6LoWPAN qui traiterait directement des paquets compressés.

Les expérimentations montrent que Mobile IPv6 est utilisable dans les réseaux 6LoWPAN même si l'en-tête de mobilité et les options associées ne sont pas compressés. Les mécanismes de compression aident légèrement à réduire la durée des transmissions des paquets IPv6 et donc la consommation énergétique et la contention sur le médium radio. Cependant, la proposition que nous avons faite s'est révélée insuffisante car il est nécessaire de faire un choix entre vitesse de détection et consommation d'énergie.

5.5 CONCLUSION

Dans ce chapitre, nous nous sommes focalisés sur une analyse expérimentale de Mobile IPv6 sur les réseaux 6LoWPAN. Les réseaux 6LoWPAN implémentent une couche d'adaptation d'IPv6 leur permettant de réduire la taille des différents entêtes utilisés dans les paquets IPv6. Ces réseaux utilisent une version modifiée de Neighbor Discovery pour s'adapter à leurs caractéristiques spécifiques. Ces modifications suppriment l'envoi périodique des messages *router advertisement* qui est utilisé par Mobile IPv6 pour détecter le mouvement du nœud mobile. Ce mécanisme n'étant plus adapté, nous avons recherché une autre méthode qui repose sur des mécanismes existants. Nous avons réalisé une implémentation qui a été faite au sein du système d'exploitation Contiki en utilisant la pile uIPv6 déjà disponible. Mobile IPv6 a été évalué en utilisant trois méthodes de compression et un total de 150 handovers de niveau 3.

Les résultats obtenus montrent que Mobile IPv6, avec de petites modifications, peut être une solution viable pour le support de la mobilité de niveau 3 dans les réseaux 6LoWPAN. A l'exception de la détection de mouvement, le reste des opérations de Mobile IPv6 sont effectuées en seulement 130.61 ms avec une compression complète des en-têtes. De plus, le mécanisme de compression peut aider à réduire la consommation

énergétique et la contention sur le médium sans fil. Cependant, le mécanisme présenté dans ce chapitre prend trop de temps pour détecter de mouvement de nœuds mobiles. De notre point de vue, il est impossible d'arriver à un compromis entre une fréquence d'émission élevée de messages *router solicitation* et une faible consommation énergétique / contention du médium. Nous sommes convaincus que la détection de mouvement devrait être effectuée par un mécanisme annexe non défini dans Mobile IPv6 ou Neighbor Discovery pour être efficace.

Dans le prochain chapitre, nous allons introduire une nouvelle solution afin d'optimiser la détection de mouvement. Notre solution se base sur la sur-écoute liée au médium radio pour détecter le voisinage du nœud mobile. Une fois que ce dernier détecte un nouveau voisin, le nœud mobile peut directement transmettre un message *router solicitation* en multicast pour initier la procédure de handover définie dans Mobile IPv6.

6

OPTIMISATION DES HANDOVERS PAR L'ÉCOUTE DU VOISINAGE

Sommaire

6.1 INTRODUCTION

Le protocole Mobile IPv6 est un protocole qui permet le support de la mobilité de niveau 3. L'objectif est de permettre aux nœuds mobiles de rester accessibles durant leur déplacement hors de leur réseau mère. Nous avons prouvé dans le chapitre 5 que le protocole Mobile IPv6 est parfaitement utilisable dans les réseaux de capteurs sans fil à l'aide de la couche d'adaptation de 6LoWPAN. Nous avons néanmoins remarqué que le mécanisme de détection de mouvement de Mobile IPv6 n'est plus applicable. Pour détecter sa mobilité, un nœud utilisant Mobile IPv6 se base sur la réception des messages *router advertisement* envoyé périodiquement par le routeur du lien. Ceci n'est plus possible dans 6LoWPAN car le protocole Neighbor Discovery a été modifié pour être adapté aux contraintes des réseaux de capteurs sans fil. Ces modifications impliquent, notamment, la suppression de l'émission périodique des *router advertisement*.

Nous avons proposé dans le chapitre 5 une solution pour pallier ce problème en nous basant sur l'un des minuteurs de

Neighbor Discovery. Cette solution n'est cependant pas entièrement satisfaisante car le mécanisme de détection de mouvement est basé sur un minuteur qui a une durée fixe définie par l'utilisateur. Le choix de cette valeur étant relativement complexe, nous pensons que la détection de mouvement doit être basée sur un mécanisme annexe afin d'être plus dynamique. Afin d'optimiser les handovers de Mobile IPv6, nous proposons de le combiner avec le protocole Mobinet, introduit dans le chapitre 4. Dans ce chapitre, nous allons présenter comment Mobinet peut optimiser les handovers de Mobile IPv6 et l'évaluation de cette combinaison de protocoles.

6.2 DÉTECTION DE MOUVEMENT BASÉ SUR L'ÉCOUTE

Afin d'optimiser la détection de mouvement de Mobile IPv6, nous proposons de combiner le protocole Mobinet, présenté dans le chapitre 4, et Mobile IPv6. Mobinet est un protocole qui permet l'écoute passive du médium radio afin de découvrir les nœuds présents dans le voisinage. Une fois que le nœud mobile entend une transmission, les informations concernant cette dernière (adresse de la source et de la destination) sont enregistrées dans une table de voisinage introduite par Mobinet.

6.2.1 *Définition de la proposition*

En analysant le changement de voisinage d'un nœud mobile, nous pouvons détecter le potentiel changement de réseau du nœud mobile. Lors de l'arrivée d'un nœud mobile dans un nouveau réseau, il va capter les transmissions de nouveaux nœuds, ce qui va modifier sa table de voisinage. Tout ajout ou retrait d'une entrée dans la table voisinage va incrémenter un compteur de changements. À intervalles réguliers, relativement courts, le nombre de changements effectués est comparé à la taille de la table de voisinage (enregistrée lors du précédent test). Si le nombre de changements dépasse un certain seuil, il est probable que le nœud mobile ait changé de réseau. Pour le vérifier, Mobinet va initier la transmission d'un message *router solicitation*, transmis en multicast. Le reste de la procédure reste identique à celle décrite dans [29]. Lors de la réception du message *router advertisement*, le nœud mobile va regarder le ou

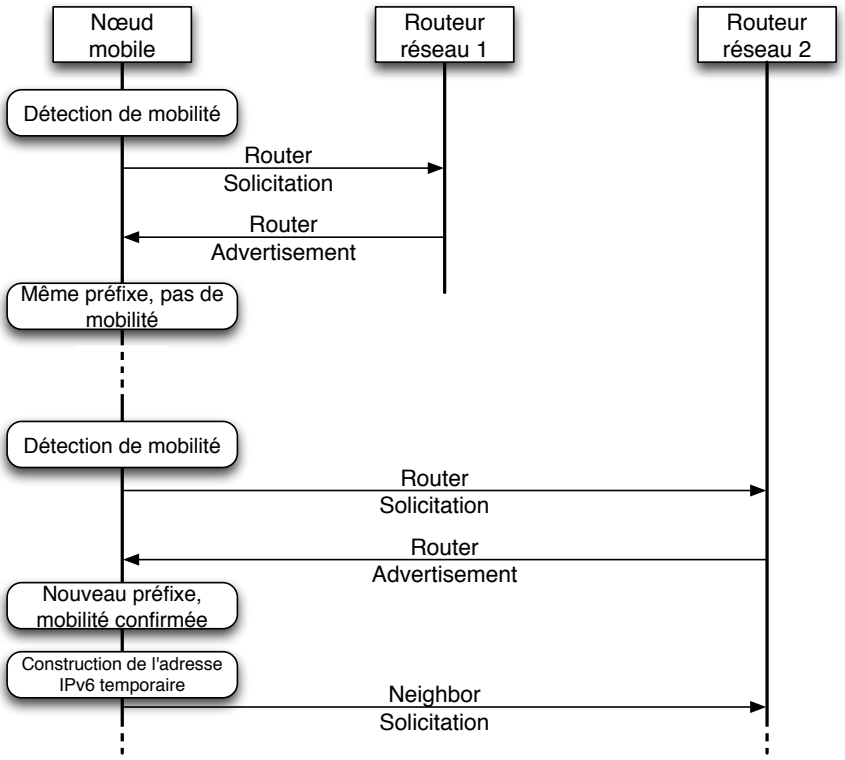


FIGURE 36: Fonctionnement de Mobinet comme mécanisme de détection de mouvement

les préfixes IPv6 annoncés. Si le préfixe actuel du nœud mobile diffère de ceux annoncés, Mobinet va donc supprimer l'adresse IPv6 temporaire et toutes les informations concernant l'ancien préfixe. Le nœud mobile construira sa nouvelle adresse IPv6 temporaire. Et une fois son enregistrement auprès du routeur effectuée avec succès, il enverra un message *binding update* vers son agent mère. La figure 36 illustre le fonctionnement de notre proposition.

Dans le cas contraire, c'est-à-dire si le préfixe IPv6 est identique à l'actuel, le nœud mobile n'est pas quitté du réseau visité et il va réinitialiser uniquement la durée de vie du routeur par défaut. Dans un routage de type mesh-under, cette procédure est suffisante car le routeur par défaut du nœud mobile est le 6LBR. Cependant dans un routage de type route-over, le routeur par défaut du nœud mobile peut également être l'un des 6LRs du réseau visité. Mobinet doit en conséquence vérifier également l'adresse source du message *router advertisement*. Si l'adresse source est différente de celle du routeur par défaut, le nœud mobile envoie un message *neighbor solicitation* afin d'en-

registrer son adresse IPv6 temporaire auprès de son nouveau routeur par défaut.

6.2.2 *Sélection du seuil de changement*

Un élément important de notre proposition est le seuil de changement de la table de voisinage de Mobinet. Sa valeur ne doit pas être choisie au hasard. Si cette valeur est trop faible, la procédure de mobilité, déclenchée trop souvent, pourrait s'avérer inutile et en conséquence augmenter la consommation énergétique du nœud mobile. A l'opposé, si cette valeur est trop élevée, le nœud mobile risque de ne jamais détecter qu'il a changé de réseau.

Pour fonctionner, le protocole Mobinet nécessite que des paquets soit échangés entre les nœuds. Dans le cas où il n'y a pas trafic dans le voisinage, plusieurs cas sont à envisager. Si la table de voisinage est vide et qu'il n'a aucun routeur par défaut enregistré, le nœud mobile détectera le réseau uniquement si une transmission a lieu. Par contre, si la table de voisinage est vide mais qu'un routeur est enregistré, le mouvement du nœud mobile peut toujours être détecté grâce au minuteur lié à la durée de vie du routeur par défaut (dont la valeur peut aller jusqu'à 150 minutes [65]). Pour finir, si la table de voisinage contient encore des entrées, ces dernières vont expirer les unes après les autres. Si les expirations sont suffisamment proches, elles peuvent dépasser le seuil de changement et lancer ainsi la procédure de mobilité comme présenté dans la section précédente. Si le seuil n'est jamais dépassé, la procédure sera automatiquement lancée dès que la dernière entrée de la table expirera.

6.3 ÉVALUATION

6.3.1 *Organisation de l'expérimentation et intégration de Mobinet*

Pour cette évaluation, nous sommes repartis de notre implémentation présentée dans le chapitre 5 et nous y avons intégré le protocole Mobinet. La nouvelle architecture logicielle est présentée dans la figure 37. La plate-forme de test ainsi que les

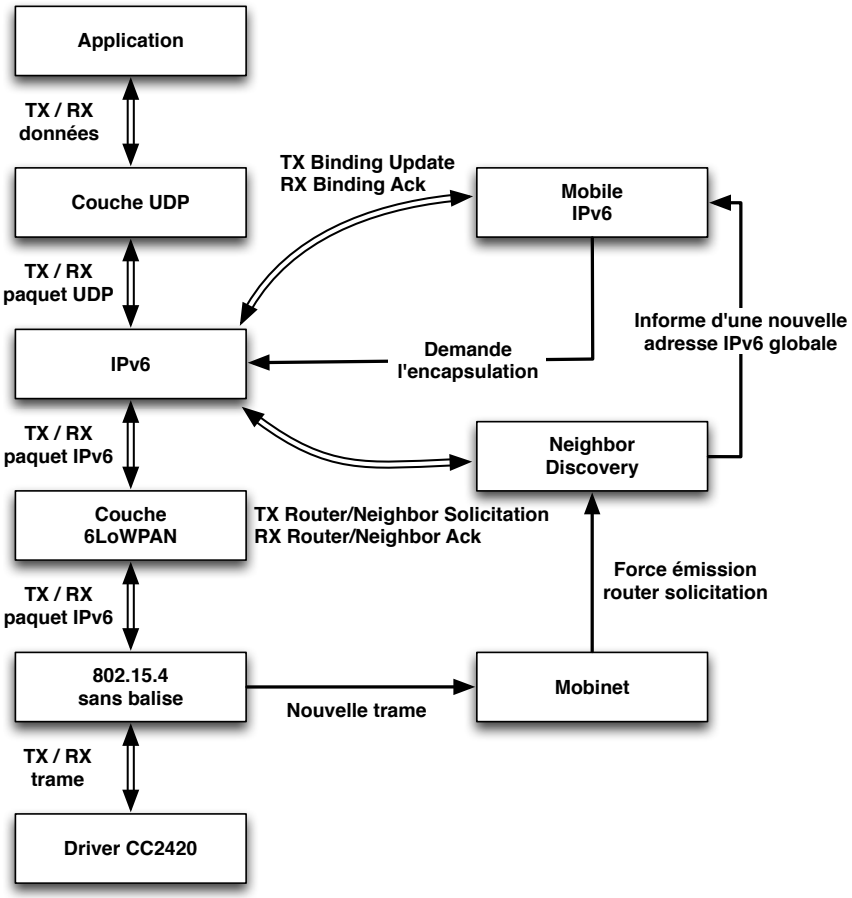


FIGURE 37: Architecture logicielle d'un nœud capteur

différents paramètres de l'expérience sont basés sur ceux présentés dans le chapitre 5. Cependant, Mobinet requiert du trafic au sein des différents réseaux visités par le nœud mobile pour être fonctionnel. Pour cela, nous avons introduit dans chaque réseau visité un nœud statique qui va utiliser la même application que le nœud mobile : il va transmettre à un nœud correspondant situé hors du réseau entre 20 et 36 octets de données toutes les secondes.

Mobinet ayant uniquement un impact sur le handover de Mobile IPv6, nous avons réduit les paramètres de l'expérience. Le nœud mobile se déplacera dans des réseaux visités constitués uniquement d'un 6LBR et du nœud fixe créant du trafic. Nous avons montré dans le chapitre précédent que le mécanisme de compression LOWPAN_IPHC, introduit dans le RFC 6282 [28], et la compression des en-têtes de Mobile IPv6 [68] permettent d'obtenir de très bon résultats que ce soit en terme de com-

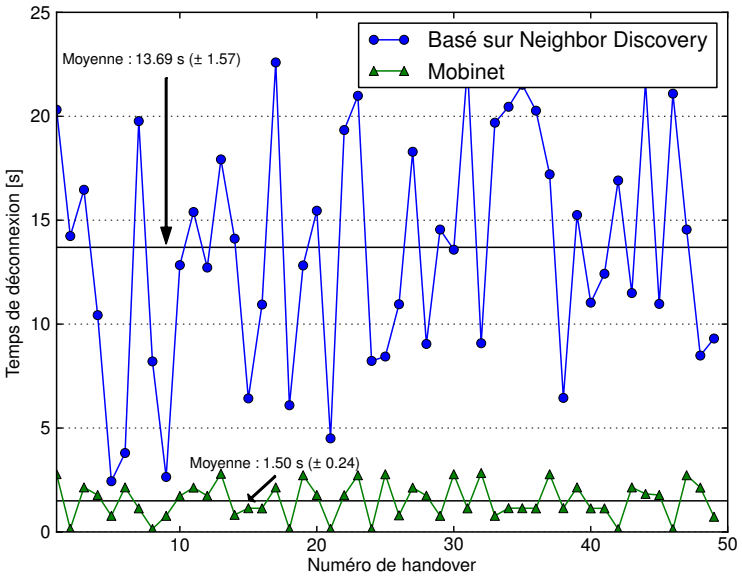


FIGURE 38: Temps de détection de mouvement avec Mobinet

pression, temps de transmission des messages de signalisation et d'économie d'énergie. Le nœud mobile utilise donc uniquement ces deux mécanismes de compression.

Le nœud mobile va utiliser le protocole Mobinet afin d'optimiser les handovers. Nous utilisons un processus d'écoute périodique avec une seconde de sommeil pour 20 ms d'écoute. Durant notre évaluation du protocole Mobinet dans le chapitre 4, ce processus d'écoute a donné les meilleurs résultats. Chaque entrée de la table de voisinage de Mobinet a une durée de vie de 10 secondes. Cette durée correspond au temps maximal mis par le nœud mobile pour traverser la zone couverte par le nœud fixe à la vitesse d'un homme qui marche. Le processus principal de Mobinet va effectuer le test sur le compteur de changements toutes les 2 secondes. Le seuil de changement a été fixé à 25% de la taille de la table de voisinage (enregistrée lors du précédent test).

6.3.2 *Détection de mouvement*

Lors de notre précédente évaluation de Mobile IPv6, nous avons montré que son mécanisme de détection de mouvement est inutilisable dans les réseaux 6LoWPAN. La figure 38 re-

présente le temps mis par le nœud mobile pour détecter le mouvement du nœud mobile. Nous avons comparé notre précédente proposition, basée sur les temporisateurs de Neighbor Discovery, avec le protocole Mobinet. Comme nous pouvons le constater, à l'aide du protocole Mobinet, le nœud mobile détecte son mouvement en moyenne au bout de $1,5$ (± 0.24) secondes. C'est 12 secondes de mieux que dans notre précédente proposition. Cette dernière reposait sur le temporisateur du routeur par défaut que nous avions fixé à 30 secondes (afin de raccourcir la durée des expériences) alors qu'elle peut aller jusqu'à 150 minutes. La mesure de la durée de détection de mouvement commence dès que le nœud mobile arrive dans le nouveau réseau et se termine quand le nœud mobile considère qu'il est déplacé : après 3 *routeur solicitation*, envoyés en unicast, sans réponse pour notre précédente solution et dès l'envoi d'un *router solicitation* en multicast pour le protocole Mobinet. Nous pouvons remarquer que Mobinet est capable de réagir très rapidement dès qu'il capture les transmissions des nœuds présents dans le nouveau réseau visité. Tout va dépendre ensuite du trafic dans la zone du nœud mobile.

6.3.3 *Impact sur la consommation énergétique*

Nos expérimentations utilisent la norme IEEE 802.15.4 en mode sans balise dans lequel la radio est toujours allumée. De ce fait, nous ne pouvons pas évaluer l'impact énergétique de Mobinet sur le nœud mobile. Cependant, nous sommes certains qu'en appliquant les différents mécanismes présentés dans le chapitre 4, nous serons capables de limiter au maximum cet impact.

Dans le cas où le nœud mobile utilise la même couche MAC mais avec le mode avec balise, Mobinet peut se baser sur son mécanisme synchronisé comme méthode d'écoute. La durée d'écoute et de sommeil se calcule à l'aide des formules suivantes (résultat exprimé en symboles) :

$$
\begin{aligned}
periode\_ecoute &= sfDuration * 2^{SO} \\
periode\_sommeil &= sfDuration * 2^{BO} - duree\_sommeil
\end{aligned}
$$

avec

$$
\begin{aligned}
0 \leqslant BO \leqslant 14 \\
0 \leqslant SO \leqslant BO \\
sfDuration &= aBaseSlotDuration * aNumSuperframeSlots \\
&= 960
\end{aligned}
$$

A 2,4 GHz et si le nœud mobile n'effectue aucune transmission, il peut avoir une période d'écoute allant de 15,36 ms (SO = 0) à environ 251,65 s (SO = 14), la période de sommeil peut aller de 0 s (SO = BO) à 251,64 s (BO = 14 et SO = 0). Comme nous pouvons le constater, ces durées vont fortement dépendre de la configuration effectuée par l'utilisateur. Cela peut aller de quelques millisecondes à environ 4 minutes. Nous avons montré dans le chapitre 4 que la consommation énergétique du nœud mobile sera accrue avec de longues périodes de sommeil. Nous conseillons donc d'utiliser des périodes de sommeil de courte durée, de 1 à 10 secondes.

Cette nouvelle évaluation a montré que l'intégration de Mobinet au sein de la pile 6LoWPAN permet d'améliorer les performances du mécanisme de détection de mouvement de Mobile IPv6.

6.4 CONCLUSION

Le principal point faible de Mobile IPv6 dans les réseaux de capteurs sans fil est son algorithme de détection de mouvement. Pour détecter son déplacement, nous avons proposé initialement d'utiliser le temporisateur de la durée de vie du routeur par défaut (provenant de Neighbor Discovery) du nœud mobile qui peut être au maximum de 150 minutes. Afin de rendre cette détection plus dynamique, nous proposons d'intégrer le protocole Mobinet dans la pile 6LoWPAN et d'utiliser son mécanisme d'écoute du voisinage pour réduire la durée du handover.

Nos expérimentations ont montré que Mobinet a un réel impact sur la vitesse à laquelle le nœud mobile détecte sa mobilité. Par rapport à nos précédentes évaluations, la durée de perte de connexion a été réduite d'un facteur supérieur à 2. Nous nous sommes également intéressés à l'impact énergétique du protocole Mobinet sur le nœud mobile. Cependant, la couche MAC utilisée ne permet pas d'évaluer l'impact de Mobinet car la radio est constamment allumée.

Dans cette partie, nous nous sommes intéressés à la gestion de la mobilité dans les réseaux de capteurs sans fil et comment la sur-écoute peut être utilisée pour améliorer les performances du nœud mobile. Dans un premier temps, nous avons proposé le protocole Mobinet dont l'objectif est le support de la mobilité au niveau 2. A l'aide de l'écoute passive du médium radio (sur-écoute), le protocole Mobinet construit une table de voisinage qui sera ensuite utilisée pour déterminer le prochain nœud vers qui transmettre les données. Cependant, la sur-écoute peut consommer beaucoup d'énergie, c'est pourquoi nous avons proposé différents processus d'écoute permettant d'en limiter l'impact.

Peu convaincus par les évaluations faites des protocoles de support de la mobilité de niveau 3 dans la littérature, nous avons effectué notre propre évaluation de Mobile IPv6 dans un environnement réaliste et utilisant les dernières versions des standards pour réseaux de capteurs sans fil. Notre évaluation a montré que Mobile IPv6 est parfaitement utilisable au sein des réseaux de capteurs sans fil. Cependant, le mécanisme de détection de mouvement de Mobile IPv6 n'est plus utilisable. Nous avons initialement proposé de le remplacer par un des temporisateurs de Neighbor Discovery, mais la durée des temporisateurs est fixe et peut être élevée. L'évaluation montre donc que ce mécanisme n'est pas idéal pour la détection de mouvement.

Nous avons finalement décidé d'utiliser le protocole Mobinet comme mécanisme de détection de mouvement. L'utilisation de Mobinet a permis d'avoir une détection plus dynamique du déplacement du nœud mobile. Cependant, la couche MAC utilisée lors de notre expérimentation ne permet pas d'économiser de l'énergie en éteignant le composant radio et donc d'évaluer l'impact de l'écoute passive de Mobinet sur la consommation énergétique du nœud mobile. Il sera donc nécessaire d'évaluer cet impact énergétique en utilisant d'autres couches MAC incluant un mécanisme d'économie d'énergie tels

que X-MAC [45] ou le mode avec balise de IEEE 802.15.4. Ce dernier dispose également de l'avantage d'intégrer un mécanisme de détection de mouvement basé sur la réception des balises émises par un nœud coordinateur. Nous pensons qu'il est encore possible d'optimiser la détection de mouvement en utilisant les données des capteurs (par exemple, faire varier le seuil de changement en fonction de la vitesse) ou en utilisant des mécanismes d'apprentissage pour optimiser la durée de vie des entrées de la table de voisinage.

Troisième partie

ÉCOUTER POUR MIEUX ÉVITER LES CONGESTIONS

7

LE PROTOCOLE CROSS-LAYER OPPORTUNISTIC MAC

Sommaire

7.1 INTRODUCTION

Durant notre étude des protocoles MAC, présentée dans le chapitre 1, nous nous sommes intéressés à la sur-écoute dont sont naturellement sujets les protocoles à préambule. Dans la seconde partie, nous avons utilisé cette *sur-écoute* pour optimiser les handovers de niveau 2 et 3. Nous nous sommes donc demandés si cette *sur-écoute* peut offrir d'autres avantages aux réseaux de capteurs sans fils.

Dans ce chapitre, nous proposons d'utiliser la *sur-écoute* pour éviter de congestions dans les réseaux de capteurs sans fil. Après avoir décrit le problème auquel nous nous attelons, nous présenterons différentes solutions de la littérature pour pallier à

ce problème dans la section 7.3. Nous présenterons notre proposition Cross-Layer Opportunistic MAC (CLOMAC) [91] qui à l'aide de la sur-écoute permet de réduire la congestion dans les réseaux de capteurs sans fil.

7.2 DESCRIPTION DU PROBLÈME

Les réseaux de capteurs actuels impliquent dans la majorité des cas un trafic de données de type convergecast (tous les paquets de données sont envoyés vers le puits). Un nœud capteur peut transmettre des paquets en fonction de 3 types de déclencheurs. La transmission de données peut être demandée par l'utilisateur du réseau via un paquet de contrôle mais elle peut également être déclenchée périodiquement par un temporisateur. L'intervalle de transmission va dépendre essentiellement de l'application utilisée. Dans le cas, par exemple, de biologgers, les données seront envoyées fréquemment (toutes les secondes par exemple). Cependant, si le réseau est déployé pour surveiller le taux d'humidité d'une ville, peu de paquets sont envoyés périodiquement au puits. Finalement, l'application peut réagir lorsqu'un événement intéressant se produit. Lors de la capture de l'événement, les nœuds capteurs localisés dans son voisinage peuvent générer un grand nombre de paquets pour surveiller plus finement cet événement.

Une telle rafale de paquets peut créer des congestions dans le réseau et, par conséquent, peut augmenter les délais de transmission et générer des pertes de paquets. En fonction de la nature de l'événement, la perte de paquets ou une latence trop élevée peut être d'une importance cruciale. Considérons l'exemple d'un réseau de capteurs sans fil chargé de surveiller les feux de forêts. Lorsqu'un départ de feu se produit, une rafale de paquets peut se produire et peut générer des congestions et des pertes d'informations. Dans le pire des scénarios, cela peut mener à des pertes matérielles et humaines. Afin de résoudre ce problème, il existe différentes techniques pour limiter les pertes et réduire le délai, telles que la synchronisation au niveau MAC [23] ou l'utilisation de fréquences multiples [92]. Il existe également des protocoles de routage opportunistes qui utilisent la sur-écoute pour router les paquets dans le réseau.

Dans ce chapitre, nous allons définir le protocole CLOMAC qui est un nouveau système d'évitement de congestion pour

réseaux de capteurs sans fil. L'objectif de CLOMAC est de bénéficier de la sur-écoute inhérente aux communications sans fil pour utiliser de manière opportuniste des chemins alternatifs vers le puits et réduire la congestion. CLOMAC est conçu pour être intégré avec les protocoles MAC à préambule [45, 44] qui sont très populaires et sujet à la sur-écoute.

7.3 ÉTAT DE L'ART

Il existe de nombreuses méthodes pour contrôler ou éviter les congestions dans les réseaux de capteurs sans fil. En particulier, le trafic convergecast favorise la génération de congestions autour du puits en raison de l'effet d'entonnoir [93] puisque tous les paquets de données sont envoyés vers le puits. Le trafic s'intensifie donc dans son voisinage et la compétition entre les nœuds du voisinage s'en trouve fortement accrue et peut générer des pertes de paquets.

7.3.1 *Protocoles MAC synchronisés et hybrides*

En synchronisant fortement les nœuds du réseau, les protocoles MAC synchronisés (c.f. chapitre 1) sont capables de résoudre ce problème. Pour envoyer ou recevoir des données de manière synchronisée, ils peuvent soit utiliser un un mécanisme basé sur des slots (IEEE 802.15.4 [23]), soit un mécanisme partagant des périodes de sommeil / activité (S-MAC [41]). En contrepartie, ces protocoles sont difficilement déployables à large échelle car établir et maintenir un telle synchronisation sur tout un réseau de capteurs est une tâche complexe.

Des protocoles MAC hybrides tels que le protocole Funneling-MAC [94] permettent de pallier à ce problème d'échelle. Ce protocole utilise un mécanisme MAC synchronisé pour les communications avec le puits et un protocole basé sur CSMA pour le reste du réseau. Chaque nœud capteur situé à un saut du puits divise sa fenêtre de communication en deux parties : une pour les communications basées sur CSMA et l'autre pour les communications synchronisées. Cependant, une telle approche hybride a plusieurs défauts. Premièrement, les paquets échangés entre les nœuds CSMA présents aux alentours de la zone synchronisée peuvent entrer en collision avec les paquets transmis

par des nœuds utilisant un mécanisme synchronisé. De plus, les nœuds hybrides sont les derniers relais pour atteindre le puits. Cette division du réseau ne fait que reporter le problème sur les nœuds à un saut du puits. Leur fenêtre de communication avec les nœuds CSMA est limitée et donc la compétition est accrue entre les nœuds CSMA pour transmettre des données vers les nœuds hybrides.

7.3.2 *Puits mobiles*

Des puits mobiles peuvent être également utilisés pour maîtriser l'effet d'entonnoir. En effet, le puits pourrait périodiquement se déplacer dans différents endroits pour équilibrer le trafic dans le réseau [95]. Cependant, chaque déplacement peut impliquer la transmission d'une grande quantité de messages de contrôle pour maintenir les chemins menant au puits et peut donc nécessiter une grande quantité d'énergie.

Le puits peut également devenir un collecteur mobile [96]. Dans une telle approche, les nœuds transmettent leur trafic vers des nœuds spécifiques qui sont chargés de stocker les données reçues jusqu'à ce que le puits soit à portée de transmission. La congestion est donc réduite puisque de nombreux nœuds de stockages peuvent être déployés dans le réseau. Mais collecter les données peut être un travail fastidieux, en particulier dans de vastes zones, car l'utilisateur doit se déplacer, lui-même, auprès de chacun des nœuds de stockage. De plus, cette méthode ne peut pas être utilisée lorsque des applications nécessitent de récupérer les données en temps réel.

7.3.3 *Plusieurs fréquences disjointes*

La plupart des protocoles de la littérature se focalisent sur les congestions se produisant autour du puits et plusieurs propositions offrent des solutions plus génériques pour éviter les congestions. Le protocole Tree-based Multi-Channel Protocol (TMCP) [92] utilise de multiples canaux radio pour augmenter le débit de transmission des paquets. TMCP divise le réseau en plusieurs arbres ayant la même racine (le puits) et alloue un canal sans fil différent à chaque arbre. Ces canaux sont sélectionnés en utilisant une propriété orthogonale : la transmission

sur un canal spécifique ne doit pas interférer avec les autres canaux. Cette méthode permet donc d'augmenter le débit global en segmentant le réseau. Cependant, allouer les différents canaux radio sans fil en limitant les interférences radio est un problème NP-complet. TMCP préconise l'utilisation d'un algorithme glouton, testant toutes les possibilités, qui peut consommer une grande quantité d'énergie lors de la création des arbres. De plus, la conception matérielle du puits n'est pas aisée car ce dernier doit être capable d'écouter sur tous les canaux au même moment.

7.3.4 *Protocoles opportunistes*

Présentation et exemple

La grande majorité des protocoles de routage multi-sauts transmettent les paquets de données vers un nœud préselectionné dans son voisinage. Cependant au sein des réseaux sans fil, les liens sont instables et peuvent donc varier au fil du temps [97] et donc le taux de paquets reçu (Packet Receive Rate (PRR)) sera variable au fil du temps en particulier pour les nœuds situés à la bordure de la zone de communication d'un nœud. Les communications sans fil impliquent potentiellement des réceptions multiples pour chaque paquet transmis et tout voisin à portée de l'émetteur obtiendra ainsi une copie du message. Originaires des réseaux mesh, les protocoles opportunistes prennent avantage de ces réceptions multiples pour retarder le choix du nœud retransmetteur, après la réception du message par les voisins. Les paquets peuvent donc naturellement prendre des chemins alternatifs pour atteindre leur destination finale.

La figure 39 illustre un exemple de transmission opportuniste. Le nœud A souhaite transmettre 5 paquets au nœud D en passant par le nœud C. Lors de la transmission, seuls les paquets 2, 3 et 5 sont reçus par le nœud C. En revanche, le nœud B a réceptionné les paquets 1 et 4 et peut donc les retransmettre vers la destination finale.

ExOR [98] est le premier protocole de routage opportuniste proposé. Avant d'émettre un paquet, un nœud sélectionne une série de nœuds localisés entre lui et la destination en fonction d'une métrique. Celle métrique utilisée ici est appelée Expected

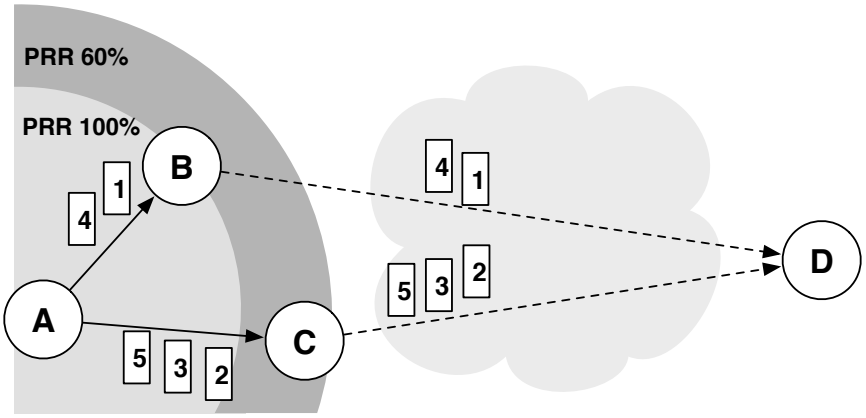


FIGURE 39: Exemple de routage opportuniste

Transmission Count (ETX) [99]. Elle mesure la qualité d'un chemin entre deux nœuds et son utilisation nécessite une connaissance de tous les liens du réseau ainsi que leur valeur ETX respective. Tout nœud contenu dans la liste incluse dans le paquet est autorisé à prévoir la retransmission du paquet. En effet, les nœuds réceptionnant le paquet vont s'organiser pour déterminer qui va le retransmettre afin d'éviter que plusieurs copies ne soient retransmises. Dans ExOR, cette sélection se fait par surécoute : un nœud détermine qu'il est sélectionné si les nœuds ayant un ETX plus faible n'ont pas retransmis ou acquitté le message avant un certain délai. Cependant, cette organisation par sur-écoute a plusieurs inconvénients : des collisions ou l'impossibilité d'entendre l'un des autres nœuds peut en effet affecter la coordination entre les nœuds et générer de la duplication. De plus, il est bien connu que les solutions centralisées ont une surcharge importante de messages en particulier lorsque la topologie est dynamique. Ce n'est donc pas une solution viable pour les réseaux de capteurs sans fil.

Protocoles opportunistes dans les réseaux de capteurs sans fil

Au sein des réseaux de capteurs sans fil, les protocoles opportunistes peuvent être regroupés en deux catégories. Les protocoles de la première catégorie se basent sur le même principe que ExOR et cherchent à calculer les meilleurs chemins vers la destination avant de transmettre le paquet tandis que les autres laissent les destinataires décider.

L'objectif des protocoles de la première catégorie est de réduire la consommation liée à la récupération de la connaissance pour effectuer le calcul des chemins. Ceci est effectué soit en utilisant uniquement une connaissance à k-sauts [100] soit en proposant des algorithmes utilisables dans un environnement distribuée [101] ou en se limitant à une seule destination, le puits [102].

Les protocoles de la seconde catégorie sont principalement des mécanismes multi-couches. Au niveau MAC, ils vont utiliser le plus souvent des informations provenant de la couche réseau pour déterminer quel voisin sera à même de retransmettre le paquet lors de la réception de ce dernier. C'est d'ailleurs le concept que nous proposons d'utiliser pour éviter les congestions au sein des réseaux de capteurs sans fil dans la section 7.4. Les protocoles de cette catégorie ont été proposés par la communauté scientifique parallèlement ou après la publication de notre proposition. Nous pouvons citer comme exemple de protocoles MAC utilisés X-MAC [103, 104], IEEE 802.15.4 [105] ou encore RI-MAC [106].

Le principal défaut de ces protocoles est la sur-écoute qui est inhérente à ces solutions. En effet, les nœuds capteurs doivent garder leur radio allumée pour récupérer un paquet de données et s'organiser pour sa retransmission. Le composant radio étant l'élément le plus gourmand en énergie, ces écoutes peuvent sérieusement réduire la durée de vie de la batterie des nœuds capteurs. Dans la prochaine section, nous détaillerons notre proposition qui tente de répondre à ce problème.

Protocoles multi-couches

7.4 CLOMAC

7.4.1 *Aperçu général*

CLOMAC (Cross-Layer Opportunistic MAC (CLOMAC)) est conçu pour éviter les congestions dans les réseaux de capteurs sans fil en utilisant les caractéristiques de leurs communications. Lorsqu'un nœud capteur souhaite envoyer un paquet, tous les nœuds présents dans sa zone de transmission peuvent réceptionner le paquet dû à la sur-écoute. De manière géné-

rale, si le nœud n'est pas le destinataire du paquet, ce paquet est simplement ignoré. Toutefois, ces réceptions multiples sont utilisées par les protocoles de routage opportunistes pour améliorer les performances du réseau. Nous proposons d'appliquer ce mécanisme aux protocoles MAC à préambule pour créer dynamiquement des chemins alternatifs afin d'atteindre le puits lorsqu'une congestion survient. Nous nous focalisons sur les protocoles MAC à préambule car ils sont sujets par nature à la sur-écoute en raison de leurs opérations asynchrones. En effet, pour s'assurer que la destinataire du paquet écoute le médium radio lors de sa transmission, un émetteur va transmettre un préambule dont la durée est supérieure à la période de sommeil d'un nœud. Ainsi tout nœud dans le voisinage de l'émetteur sera éveillé pour la transmission du paquet de données (voir chapitre 1). Parmi ces protocoles, certains incluent des informations telles que la source ou la destination dans le préambule. Dans la suite, nous considérerons un protocole MAC à préambule générique dont le préambule ne contient pas de telles informations.

Détection de congestions

CLOMAC est conçu pour opérer avec un trafic de type convergecast (tous les paquets de données sont envoyer vers le puits) dans lesquels les paquets de données incluent un numéro de séquence. Pour détecter une congestion, CLOMAC se repose sur le mécanisme de retransmission inclus dans la majorité des protocoles MAC. De manière générale, un émetteur continue de retransmettre un paquet tant qu'il n'a pas obtenu de confirmation de sa bonne réception (acquittement (ACK)) ou tant qu'il n'a pas atteint le nombre maximal de retransmissions. Dans CLOMAC, un émetteur est considéré comme congestionné lorsque le nombre de retransmissions atteint un certain seuil, que nous appelons Congestion Threshold (CT). Pour informer les nœuds présents dans le voisinage, nous avons inclus le nombre de retransmissions dans l'en-tête MAC des paquets. Un nœud entendant un tel paquet peut donc déterminer si l'émetteur est congestionné en comparant le CT avec le nombre actuel de retransmissions.

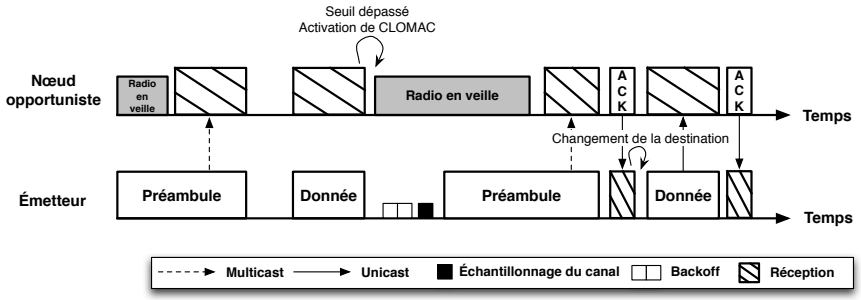


FIGURE 40: Création de chemins alternatifs avec CLOMAC

Construction de chemins alternatifs

Pour réduire les congestions, CLOMAC se base sur le même principe que les protocoles de routage opportunistes. Lorsqu'un nœud détecte un émetteur congestionné, il accepte le paquet que l'émetteur tente de retransmettre à la place du destinataire initial. Ces nœuds sont nommés nœuds opportunistes dans le reste du chapitre. Une fois qu'un nœud opportuniste a entendu un paquet dont le nombre de retransmissions a dépassé le CT, il s'endort à nouveau en attendant la prochaine transmission de l'émetteur. A la réception du prochain préambule, le nœud opportuniste envoie en réponse un acquittement indiquant qu'il demande le paquet arrivant. Si l'émetteur reçoit un acquittement après son préambule, il change le destinataire du paquet pour le nœud opportuniste. Dès réception, le nœud opportuniste acquitte le paquet de données et le retransmet vers son prochain saut. Une telle procédure permet de s'assurer que le nœud opportuniste ne retransmet pas un paquet déjà acquitté par un autre nœud. Par exemple, un nœud opportuniste peut ne pas être à portée de la destination réelle du paquet et donc ne peut pas entendre son acquittement. Toute cette procédure est illustrée dans la figure 40.

Cette procédure a également l'avantage de synchroniser les nœuds opportunistes potentiellement présents dans le voisinage de l'émetteur. Même si de multiples nœuds opportunistes tentent d'acquitter le préambule, seul le destinataire du paquet de données est autorisée à le retransmettre. Par ce moyen, une seule copie de la donnée sera finalement reçue par le puits.

Synchronisation entre nœuds opportunistes

Le problème principal des protocoles de routage opportunistes est la synchronisation entre les nœuds pour s'assurer qu'une seule et unique copie de la donnée soit retransmise vers la destination. Aussi, les nœuds opportunistes ne doivent pas créer de boucles dans le réseau. Pour sélectionner un nœud opportuniste, CLOMAC utilise une heuristique basée sur les informations du protocole de routage. Un mécanisme simple est d'autoriser un nœud à être opportuniste seulement si ce dernier est plus proche du puits que l'émetteur, en fonction de la métrique utilisée par le protocole de routage. Un tel mécanisme s'assure que CLOMAC ne crée aucune boucle de routage mais peut limiter le nombre de nœuds opportunistes dans le voisinage d'un émetteur et donc potentiellement réduire le nombre de chemins alternatifs disponibles.

La solution à ce problème est d'autoriser les nœuds ayant une métrique de routage équivalente à celle du nœud émetteur, à devenir opportunistes. Cependant, router des paquets entre de tels nœuds peut créer des boucles de routage dans le réseau. En effet, si le paquet est continuellement capturé par des nœuds ayant la même métrique, il ne s'approchera jamais de sa destination finale et restera indéfiniment dans le réseau. Pour éviter cela, CLOMAC enregistre les informations concernant les paquets émis dans une table distribuée sur chaque nœud. La source et le numéro de séquence de tout paquet émis ou retransmis par un nœud sont enregistrés dans cette table durant une courte période. Un nœud accepte d'être opportuniste pour un paquet seulement si l'émetteur et le numéro de séquence ne sont pas déjà inscrit dans cette table. Par ce moyen, un paquet ne peut pas être transmis plusieurs fois par le même nœud.

Par la suite, nous allons détailler l'intégration de CLOMAC dans le protocole B-MAC [44]. Avec B-MAC, un nœud doit écouter le préambule plus le paquet de données pour déterminer la destination du paquet de données. Cette sur-écoute est l'un des principaux inconvénients de B-MAC mais peut devenir un avantage avec CLOMAC.

7.4.2 *Intégration dans le protocole B-MAC*

Le protocole B-MAC [44] est l'un des travaux déterminants dans les protocoles MAC à préambule. B-MAC utilise un long préambule suivi d'un court mot de synchronisation (SYNC) avant tout envoi de données. Le paquet de données est envoyé directement après le mot SYNC. Comme ni le préambule, ni le mot de synchronisation ne contiennent des informations sur le destinataire, tous les nœuds à portée de transmission doivent rester en mode réception jusqu'à la réception du paquet de donnée. En conséquence, CLOMAC va pouvoir profiter de la surécoute qui est inhérente à B-MAC. Nous avons cependant effectué quelques modifications à B-MAC pour y intégrer CLOMAC. Nous avons, dans un premier temps, inclus un petit intervalle (Inter-Frame Space (IFS)) entre le mot de synchronisation et le paquet de données pour permettre aux nœuds opportunistes d'acquitter le préambule et devenir le nouveau destinataire du paquet. Ensuite, nous avons inclus dans l'en-tête MAC le numéro de séquence, l'émetteur initial et le numéro de retransmission pour le paquet comme requis par CLOMAC. Cette nouvelle version de B-MAC combinée avec CLOMAC est appelée B-MAC + CLOMAC dans le reste de ce chapitre.

7.4.3 *Cas d'erreurs*

Nous avons identifié plusieurs cas d'erreurs lors de l'utilisation de B-MAC + CLOMAC. Les sections suivantes décrivent brièvement les problèmes potentiels et comment CLOMAC les gère.

Collisions entre les acquittements de préambules

Avant d'envoyer un acquittement de préambule, un nœud opportuniste vérifie si le médium radio est libre pour éviter les collisions avec d'autres communications. En particulier, un autre nœud pourrait également envoyer un acquittement pour le même préambule. Cependant, une collision peut également se produire au niveau de l'émetteur si deux nœuds opportunistes, pas à portée radio l'un de l'autre, envoient un acquittement en même temps. Dans CLOMAC, les nœuds opportunistes détectent une telle collision si le destinataire du paquet

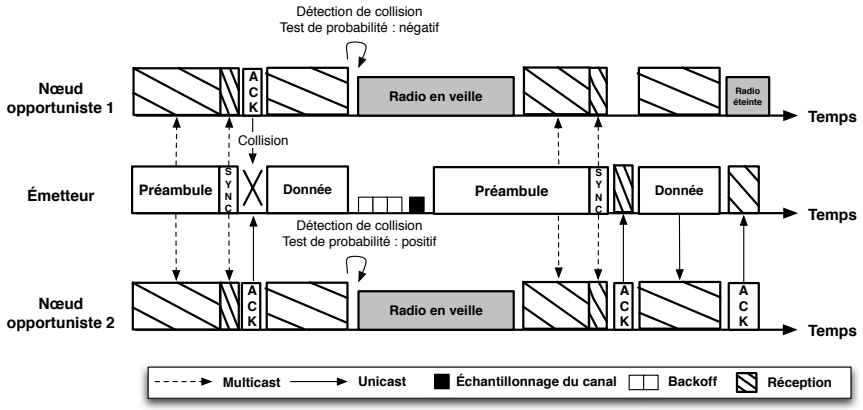


FIGURE 41: Détection de collisions entre acquittements de préambules

de données en attente n'a pas été changé : la source n'a donc pas pris en compte leurs acquittements. Pour éviter d'autres collisions entre des acquittements de préambule, les nœuds opportunistes peuvent envoyer à nouveau un acquittement lors de la prochaine transmission avec une probabilité p. Après une collision entre acquittements de préambules, chaque nœud opportuniste va tirer un nombre aléatoire compris dans l'intervalle $[0, 1]$. Si ce nombre est plus grand ou égal à p, il peut envoyer un nouvel acquittement au prochain préambule. Ce scénario est représenté dans la figure 41.

Acquitter le bon paquet de données

Comme le préambule et le mot de synchronisation de B-MAC ne contiennent aucune information sur l'émetteur ou le destinataire, un nœud opportuniste peut acquitter le préambule envoyé par un autre émetteur. De plus, un nœud opportuniste peut acquitter un paquet de données qui a déjà été reçu avec succès par le destinataire (la retransmission d'un tel paquet est la conséquence d'une collision lors de la réception de l'acquittement). Pour s'assurer que le préambule acquitté est bien celui du bon émetteur du paquet de données, les nœuds opportunistes incluent la source et le numéro de séquence utilisés dans l'en-tête MAC de chaque paquet (comme requis par CLOMAC). Lors de l'acquittement d'un préambule, un nœud opportuniste inclut la source et le numéro de séquence pour lequel il souhaite être opportuniste. Un émetteur accepte un acquittement de préambule uniquement si la source et le numéro de séquence inclus correspondent au paquet en attente de trans-

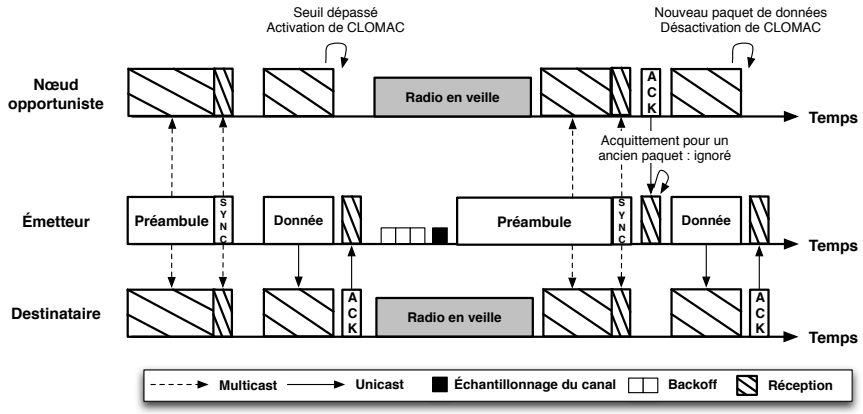


FIGURE 42: Acquitter le préambule du bon paquet de données

mission. Dans le cas contraire, l'émetteur ignore simplement l'acquittement de préambule et transmet le paquet en attente à la destination originelle. Quand un nœud opportuniste réceptionne un nouveau paquet après la transmission d'un acquittement de préambule, il s'endort à nouveau et attend la prochaine transmission. Ce scénario est illustré dans la figure 42.

7.5 ÉVALUATION DE NOTRE PROPOSITION

Nous avons intégré CLOMAC dans la version de B-MAC disponible dans le simulateur WSNet [80]. Ce simulateur inclut des modèles de propagation et d'interférence radio variés qui facilitent l'étude des couches basses de la pile de protocoles.

En plus du développement de B-MAC + CLOMAC, nous avons également implémenté un mécanisme pour détecter la duplication de paquets. Ce processus utilise les numéros de séquence des paquets qui sont inclus dans l'en-tête MAC. Si un nœud tente de transmettre un message qui a déjà été reçu avec succès, le destinataire rejette cette copie mais envoie quand même un acquittement pour arrêter le processus de retransmission. Une telle situation se produit quand l'acquittement d'un paquet de données n'est pas reçu par l'émetteur. Ce mécanisme est inclus respectivement avec B-MAC (appelé *B-MAC sans duplications*) et B-MAC + CLOMAC (appelé *B-MAC + CLOMAC sans duplications*). Dans la suite, nous comparons les performances de toutes ces solutions (i.e. B-MAC, B-MAC + CLOMAC et leurs versions sans duplication) en terme de paquets perdus et de consommation énergétique.

Paramètres simulés	Valeurs
Topologie	Grille carrée (100 m × 100 m) de 100 nœuds capteurs
Période d'envoi des données	Lors de la détection d'un événement
Étendue d'un événement	15 m
Nombre d'événements	5, 10, 15, 20, 25, 30 événements
Modèle de routage	Protocole de routage basé sur un gradient
Protocole MAC	B-MAC, B-MAC + CLOMAC Durée du préambule 100 ms
Nombre maximal de retransmissions	8
Modèle radio	Fréquence : 868 MHz BPSK Porté : 15 m
Modèle énergétique avec une batterie de 3V	Idle : 1.6 mA, RX : 14.6 mA, TX : 16.4 mA, Init. radio : 8.2 mA
Durée et nombre de simulations	600 secondes simulée 50 fois
Paramètres de CLOMAC	**Valeurs**
Congestion Threshold (CT)	4
Durée de l'IFS	1 ms
Probabilité d'acquittement, p	1/2

TABLE 8: Paramètres utilisés dans nos simulations.

7.5.1 *Environnement de simulation*

Notre scénario simulé consiste en 100 nœuds capteurs sans fil déployés en une grille carré de 100 m × 100 m. Chaque nœud a une portée de transmission de 15 mètres et a donc en moyenne 8 voisins. Le réseau utilise un protocole de routage basé sur un gradient [81] pour calculer le prochain saut en direction du puits. Chaque nœud capteur utilise une application basée sur les événements : les données sont transmises uniquement lorsqu'un événement est détecté. Au début de la simulation, nous avons réparti tous les événements sur les 600 secondes

de simulation (divisées en slots d'une seconde) en utilisant un processus de Poisson :

$$\lambda_{poisson} = \frac{evt}{600} \quad | \quad evt \in 5, 10, 15, 20, 25, 30$$

La localisation des événements est distribuée uniformément dans la zone. Lors de la détection d'un événement, un nœud envoie des paquets de données de 4 octets en direction du puits durant toute la longueur de l'événement. Un événement dure 10 secondes. Les nœuds utilisent alternativement B-MAC ou B-MAC + CLOMAC avec une durée de préambule fixée à 100 ms. Le modèle énergétique émule la consommation d'un composant émetteur / récepteur CC1101 [37]. Les valeurs de consommation et tous les paramètres de simulation sont reportés dans la table 8.

L'heuristique utilisée pour sélectionner un nœud opportuniste est basée sur le rang du nœud. Dans les protocoles de routages basés sur un gradient, le rang fait référence à la distance (nombre de sauts) entre le nœud et le puits. Dans CLOMAC, seul les nœuds avec un nombre de saut inférieur ou égal au rang de l'émetteur peuvent devenir opportunistes pour cet émetteur. Un nœud opportuniste considère un émetteur comme congestionné si ce dernier retransmet plus de 4 fois le même paquet de données. Nous avons sélectionné un nombre de retransmissions et un CT bas (respectivement 8 et 4) pour montrer qu'il n'est pas nécessaire d'avoir un grand nombre de retransmissions pour obtenir de bons résultats (par exemple, le protocole CTP [107] utilise par défaut 32 retransmissions). Si une collision se produit durant un acquittement de préambule, les nœuds ont une probabilité de 1/2 pour acquitter le prochain préambule, ainsi les chances de collisions futures sont réduites.

7.5.2 *Résultats et analyse*

Chaque couple (*protocole MAC*, *nombre d'événements*) a été simulé 50 fois. Les simulations incluent une période d'initialisation du réseau de 60 secondes. Chaque jeu de simulations utilise la même graine de génération de nombres aléatoires pour évaluer les différentes solutions dans les mêmes conditions. Les résultats présentés dans cette section sont une moyenne de toutes les données collectées durant les simulations. L'intervalle de confiance à 95% indique la fiabilité de nos mesures.

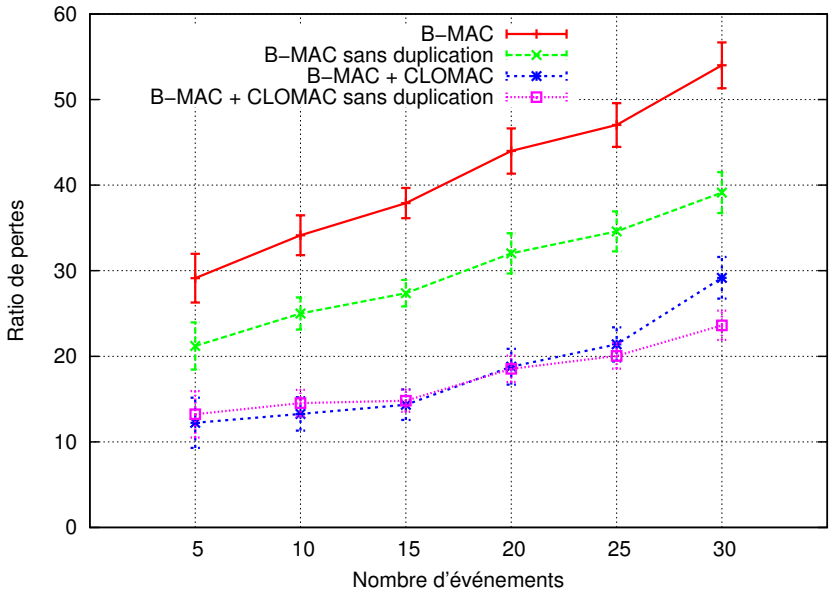


FIGURE 43: Ratio de perte de paquets au niveau applicatif

Pourcentage de paquets perdus

La figure 43 représente le pourcentage de paquets perdus au niveau applicatif. Un paquet est considéré comme perdu pour l'application lorsque le nombre de retransmission au niveau MAC a atteint la valeur maximale autorisée (fixée à 8 dans nos simulations). Quelque soit le protocole utilisé, les nœuds ont envoyé entre 249 (±18) paquets avec 5 événements et 1488 (±42) paquets avec 30 événements. Les résultats montrent que l'intégration de CLOMAC dans B-MAC surpasse B-MAC et réduit de 45% à 55% le nombre de paquets perdus. CLOMAC rend cela possible en limitant les congestions et par conséquent, les paquets perdus en créant de manière passive des chemins alternatifs.

Afin d'analyser plus finement les pertes de paquets, nous avons reporté les raisons de pertes de paquets au niveau MAC dans la table 9. Un paquet est considéré comme perdu pour la couche MAC dès lors que l'émetteur programme sa retransmission. Avec B-MAC, nous pouvons observer que 5542 paquets (sur 15455 émis au niveau MAC) sont perdus en moyenne parce qu'il *ne sont pas capturés* par l'émetteur radio. Cela signifie que l'émetteur radio s'est accroché sur un autre signal (un paquet de données ou un préambule par exemple) car la sensibilité lors de la réception était meilleure. Cela peut se produire lors

Protocole MAC	Paquets transmis	Paquets perdus		Localisation (près de)		
		erreur	non capturé	puits	événement	autre
B-MAC	15455 (± 694)	6332 (± 231)	5542 (± 160)	144	2542	13879
B-MAC sans duplication	10424 (± 531)	4123 (± 114)	3403 (± 81)	74	2473	8557
B-MAC + CLOMAC	11000 (± 669)	6177 (± 104)	1217 (± 72)	80	1967	9295
B-MAC + CLOMAC sans duplication	8504 (± 446)	4572 (± 63)	955 (± 47)	56	1925	6752

TABLE 9: Nombre moyen de paquets perdus au niveau MAC et leurs localisations dans le réseau avec 30 événements

de la capture d'un événement car de nombreux émetteurs sont regroupés dans une petite zone. B-MAC + CLOMAC est capable de limiter ces pertes en utilisant des nœuds opportunistes pour contourner les zones congestionnées. Cependant, la majorité des pertes de paquets sont générées par des *erreurs du paquet de données*. Le paquet de données est bien capturé par la radio mais il est rejeté car le paquet est corrompu. Cela peut provenir des interférences environnementales ou lorsque deux paquets entrent en collision. Comme B-MAC n'utilise pas de mécanisme tel que RTS/CTS (voir chapitre 1), il est sujet au problème du nœud caché [38]. CLOMAC et B-MAC ont des résultats similaires avec cette erreur : la majorité des pertes de paquets se produit durant la première phase du processus de retransmission avant que CLOMAC ne soit activé.

En examinant en détail la localisation des pertes de paquets, nous pouvons observer que la plupart d'entre eux sont perdus hors de la zone de l'événement. Avec la valeur du CT utilisée dans les simulations, CLOMAC est activé dans des zones fortement congestionnées (les zones où les événements se produisent) mais également dans d'autres parties du réseau tel que la dorsale de routage.

Duplication de paquets

Lorsqu'une collision survient durant la transmission d'un acquittement, l'émetteur assume que le paquet envoyé a été perdu et programme sa retransmission. En conséquence, une destination peut réceptionner des copies multiples du même paquet.

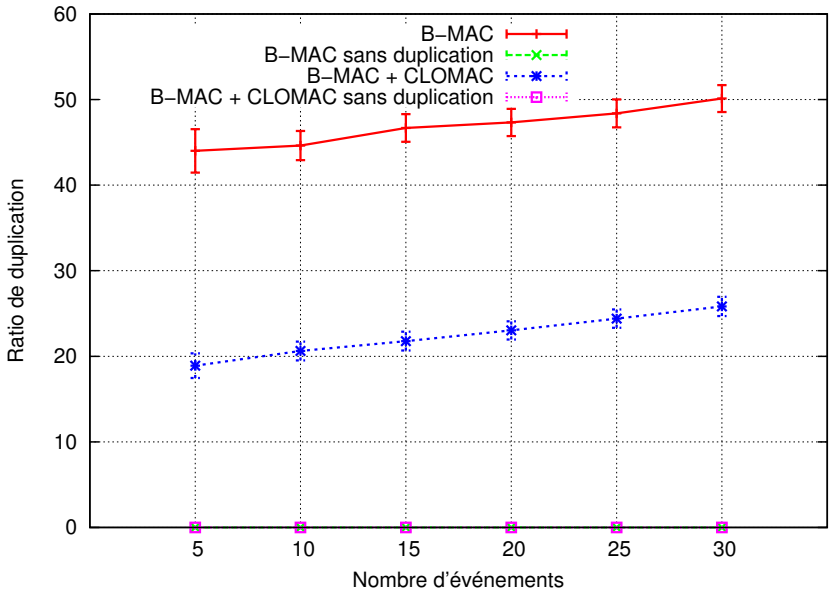


FIGURE 44: Pourcentage de paquets dupliqués reçus par le puits

Cependant, la transmission de multiples copies du même paquet vers le puits consomme des ressources, comme de l'énergie, en vain. La figure 44 représente le pourcentage de paquets dupliqués qui ont été reçus par le puits à la fin de la simulation. Les résultats montrent qu'en plus de réduire les pertes de paquets, CLOMAC aide également à réduire le nombre de paquets dupliqués. En distribuant les paquets sur de multiples chemins, CLOMAC réduit le niveau de contention sur le médium radio et donc les chances de collisions. Par conséquent, l'introduction de numéro de séquence dans l'en-tête MAC peut être une solution pratique pour limiter la duplication de paquets dans les réseaux de capteurs sans fil.

CLOMAC réduit les congestions en utilisant des chemins alternatifs pour router les paquets vers le puits. Ces chemins additionnels sont créés de manière opportuniste et ne nécessitent donc aucun message de contrôle, comme expliqué dans la section 7.4.1.

La figure 45 représente le nombre de routes utilisées pour atteindre le puits. Un chemin est composé d'une séquence unique de sauts pour atteindre le puits. Avec seulement 5 événements dans le réseau, CLOMAC augmente déjà le nombre de chemins utilisés d'un facteur 2 (34 chemins sont utilisés par B-MAC + CLOMAC et seulement 17 par B-MAC). Avec 30 événements dans le réseau, le nombre de chemins utilisés atteint 160 pour

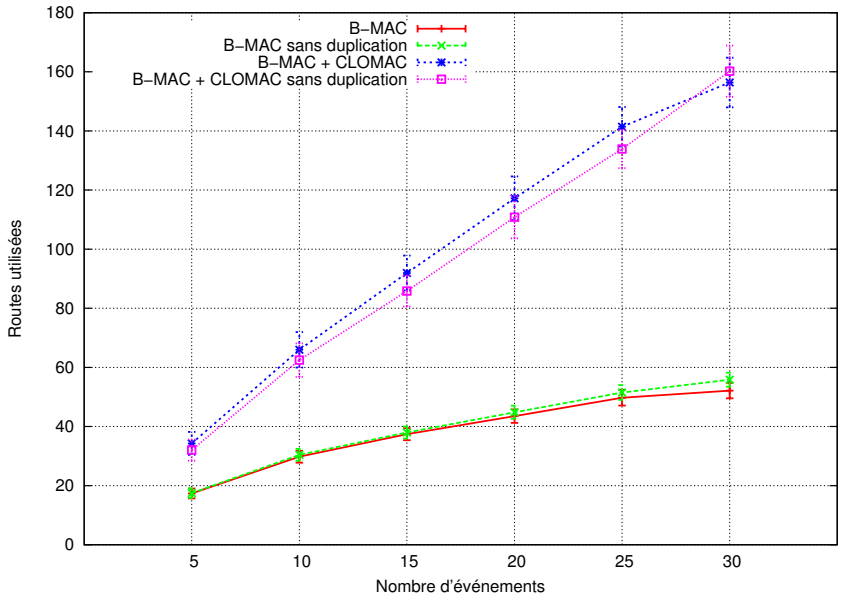


FIGURE 45: Nombre de chemins utilisés pour atteindre le puits

B-MAC + CLOMAC. En effet, plus le nombre d'événements est grand, plus le nombre de nœuds opportunistes croît, par conséquent, plus de chemins sont créés et utilisés. A l'opposé, B-MAC est seulement capable d'utiliser moins de 60 chemins (chemins calculés par le protocole de routage). Il est important de noter que le nombre de chemins augmente avec le nombre d'événements car de nouvelles sources de messages apparaissent avec ces événements. Cette observation est plus claire dans la figure 46 qui représente les liens utilisés dans le réseau. Les liens utilisés par à la fois par B-MAC et B-MAC + CLOMAC sont représentés par des traits pleins et les liens additionnels utilisés par B-MAC + CLOMAC par des traits en pointillés. Les cercles pleins représentent les zones des événements qui sont apparus durant les simulations. Nous pouvons remarquer que plus de liens sont utilisés par B-MAC + CLOMAC (108 liens) que B-MAC (42 liens) pour retransmettre les paquets. Une plus forte proportion du réseau est donc utilisée. Cela a pour résultat de réduire la congestion et de distribuer plus équitablement la consommation énergétique requise pour délivrer tous les paquets de données au puits.

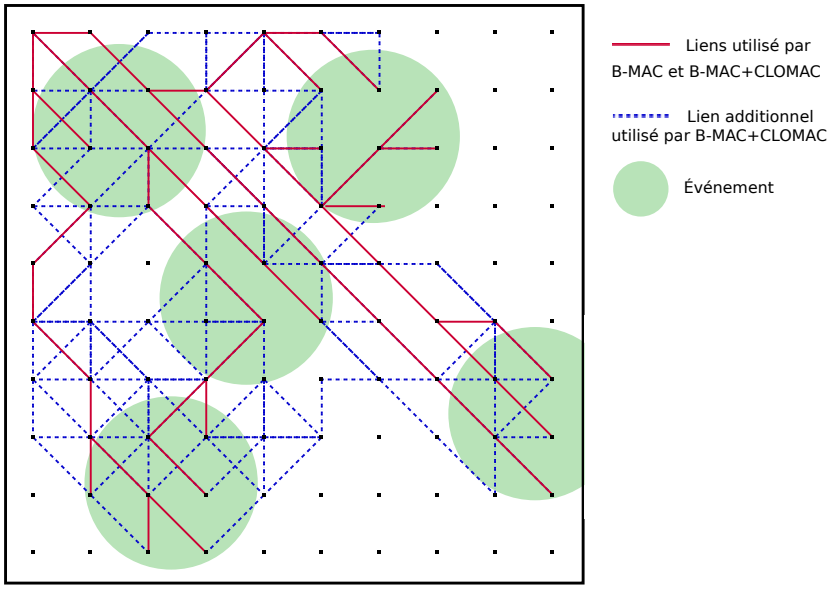


FIGURE 46: Chemins utilisés par les paquets pour atteindre le puits avec 5 événements.

Étude de la consommation énergétique

Cette dernière observation est analysée plus en détail dans la figure 47 qui représente la consommation énergétique du réseau de capteurs sans fil à la fin de la simulation. Les mécanismes opportunistes sont basés sur la sur-écoute, ce qui rend l'analyse de leur coût énergétique d'une importance cruciale pour les réseaux de capteurs sans fil. Malgré l'augmentation du temps d'écoute par l'ajout de l'intervalle (IFS) entre le mot de synchronisation et le paquet de données, nous pouvons observer B-MAC + CLOMAC est capable de réduire significativement la consommation énergétique des nœuds capteurs. La sur-écoute nécessaire pour faire fonctionner CLOMAC est déjà présente dans B-MAC et donc seule l'IFS et l'acquittement de préambules sont des sources additionnelles de consommation énergétique. Cependant ce coût est largement compensé par la réduction du nombre de retransmissions et des paquets dupliqués qui peuvent apparaître dans B-MAC seul. Nous pouvons également remarquer que la consommation énergétique de B-MAC + CLOMAC est pratiquement la même que dans sa version sans duplications. Comme expliqué dans la section 7.5.1, le destinataire doit acquitter le paquet (même si ce dernier a déjà été reçu) pour arrêter le processus de retransmissions. En conséquence, seule l'énergie utilisée pour transmettre les pa-

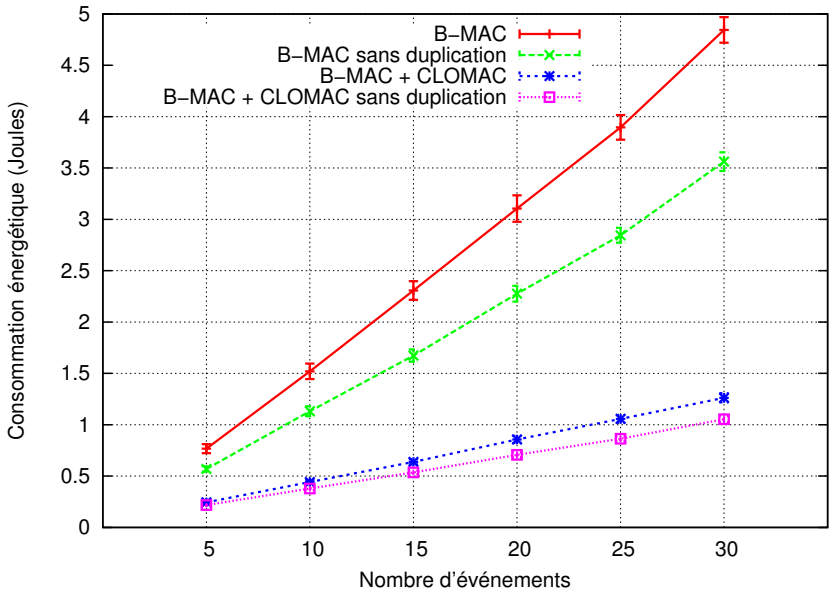


FIGURE 47: Consommation énergétique moyenne des nœuds capteurs dans le réseau

quets dupliqués est économisée en comparaison à B-MAC + CLOMAC seul.

7.6 CONCLUSION

Dans ce chapitre, nous nous sommes concentrés sur l'évitement de congestions dans les réseaux de capteurs sans fil. Lors de l'utilisation d'applications surveillant des événements, des rafales de paquets peuvent être transmises lors de la détection d'un événement pouvant créer des congestions. Nous avons proposé la solution CLOMAC (Cross-Layer Opportunistic MAC) qui est basée sur la sur-écoute naturelle des protocoles MAC à préambule. L'objectif de CLOMAC est de créer des chemins alternatifs vers le puits. Une fois que les voisins d'un émetteur ont détecté une congestion (détection basée sur le nombre de retransmissions du même paquet de données), ils peuvent accepter le paquet à la place du destinataire initial. La détection de la congestion est basée sur le nombre de retransmissions du même paquet de données. Si ce nombre de retransmissions dépasse un seuil (CT) que nous avons défini, l'émetteur est considéré comme congestionné. Ceci permet d'éviter que CLOMAC s'active inutilement. Un activation trop précoce

va augmenter la durée d'écoute du canal par les nœuds et donc accroître leur consommation énergétique. CLOMAC inclut également un mécanisme de synchronisation entre les nœuds opportunistes de manière à ne retransmettre qu'une unique copie de chaque paquet. Afin de limiter les congestions, CLOMAC va créer des chemins alternatifs de manière passive et ainsi soulager les chemins de routage principaux. CLOMAC peut être adapté aux protocoles MAC à préambule existants.

Pour ce premier travail, nous avons intégré CLOMAC dans B-MAC [44]. La combinaison de ces deux protocoles est nommée B-MAC + CLOMAC. Les résultats des simulations ont montré que B-MAC + CLOMAC est bien plus efficace que le protocole B-MAC pour délivrer les paquets au puits. B-MAC + CLOMAC augmente significativement le nombre de chemins pour atteindre le puits. Cela réduit le niveau de congestion sur le médium radio ce qui, par conséquent, réduit drastiquement les pertes de paquets. Comme ces chemins alternatifs sont créés de manière opportuniste, ils ne requièrent aucun message de contrôle supplémentaire pour être maintenus. Les solutions opportunistes souffrent généralement d'une forte consommation énergétique due à la sur-écoute. Cependant, B-MAC est déjà sujet à ce phénomène et donc CLOMAC n'implique aucune surconsommation pour son fonctionnement. De plus, la consommation énergétique globale est réduite drastiquement avec B-MAC + CLOMAC par rapport à B-MAC car moins de paquets ont été envoyés. Nous avons pourtant observé que le seuil de congestion (CT), utilisé pour activer CLOMAC, devrait être plus dynamique pour réagir plus efficacement lors des congestions et ainsi réduire encore les pertes de paquets.

CONCLUSION GÉNÉRALE

CONCLUSION GÉNÉRALE ET PERSPECTIVES

L'observation de l'environnement, de la vie animale et surveillance médicale à distance sont quelques exemples d'applications offertes par les réseaux de capteurs sans fil. Ces réseaux, en raison de leurs fortes contraintes (énergie, mémoire et puissance de calcul), soulèvent de nouveaux défis dans le domaine des communications radio. De plus en plus d'applications utilisent des capteurs placés sur des éléments mobiles pour améliorer la couverture du réseau et obtenir des résultats plus fins et plus rapidement. Un nœud mobile peut se déplacer au sein du réseau dans lequel il a été déployé initialement ou, tout comme les équipements des réseaux IP, à travers de multiples réseaux.

La gestion de cette mobilité va s'effectuer sur plusieurs niveaux. Au niveau MAC, il est nécessaire qu'un nœud mobile puisse être capable de s'insérer dans les communications. De nombreuses solutions ont été proposées pour pallier ce problème mais ces dernières reposent sur un mécanisme de niveau supérieur pour déterminer vers quel nœud présent dans son voisinage il doit transmettre ses données. Il est cependant fort peu probable que le mécanisme utilisé par un nœud mobile puisse fonctionner avec celui du réseau dans lequel il se trouve, l'empêchant ainsi de pouvoir déterminer son prochain saut.

L'intégration du protocole IPv6 permet aux réseaux de capteurs sans fil de bénéficier de plusieurs avantages tels que la réutilisation de structures existantes ou l'interconnexion des réseaux sans nécessiter d'entités intermédiaires (par exemple un serveur proxy). Tout comme dans les réseaux IP, la mobilité d'un nœud pose problème lorsque ce dernier change d'adresse car les sessions entre un nœud mobile et ses correspondants seront réinitialisées. Pour maintenir cette connectivité, il est donc nécessaire d'utiliser un protocole de support de la mobilité fonctionnant au niveau 3. Plusieurs protocoles de support de la mobilité ont été standardisés pour les réseaux IP et la communauté scientifique s'en est inspirée pour proposer de nouvelles solutions sans pour autant évaluer si ces protocoles sont utilisables au sein des réseaux de capteurs sans fil.

CONTRIBUTIONS DE LA THÈSE

Durant l'étude des protocoles MAC, nous nous sommes particulièrement intéressés à la sur-écoute liée au médium radio sans fil et comment cette sur-écoute peut être utilisée pour améliorer les performances des réseaux de capteurs sans fil.

Gestion de la mobilité

À partir de la sur-écoute, notre première contribution, le protocole Mobinet, est capable de découvrir les nœuds présents dans le voisinage d'un nœud mobile. En utilisant ces informations, Mobinet permet de déterminer vers quel voisin le nœud mobile peut transmettre ses données. Cette sélection peut être effectuée de manière aléatoire ou de manière plus fine. En étudiant les flux de données, le nœud mobile peut déterminer le meilleur voisin selon la métrique utilisée par le protocole de routage du réseau visité. Cependant, la construction de la table de voisinage va consommer beaucoup d'énergie. Pour limiter cette consommation, nous avons introduit différents processus d'écoute avec pour objectif de mettre la radio en veille le plus souvent possible. Le premier processus cherchera à découvrir le voisinage uniquement lorsqu'un paquet doit être envoyé. Le second processus cherchera toujours à avoir une entrée dans la table de voisinage en écoutant le médium radio lorsque celle-ci est vide. Finalement, le dernier processus est inspiré des protocoles MAC à préambule et va alterner des périodes d'écoute et de sommeil.

Nous avons évalué les performances de Mobinet et de ses processus d'écoute par simulation. Cette évaluation a mis en évidence que le protocole Mobinet permet aux nœuds mobiles de transmettre efficacement leurs paquets au réseau visité tout en limitant l'impact énergétique sur ce dernier. Notre évaluation a également montré qu'un processus par alternance avec des périodes de sommeil et d'écoute de respectivement 1 s et 20 ms permettent de limiter efficacement la consommation énergétique d'un nœud mobile.

Nous nous sommes ensuite intéressés au support la mobilité de niveau 3. Peu convaincus par les évaluations faites du protocole Mobile IPv6, nous avons mené nos propres expérimen-

tations. Ces expérimentations ont été effectuées sur une plateforme déployée dans nos locaux. Notre implémentation, réalisée sur le système d'exploitation Contiki, utilise les derniers standards en vigueur pour les réseaux 6LoWPAN. Cette évaluation a mis en évidence que : Mobile IPv6 est parfaitement utilisable dans les réseaux de capteurs sans fil mais que son mécanisme de détection de mouvement basé sur les messages *router advertisement* ne l'est pas.

Nous avons donc proposé une première solution basée sur les temporisateurs de Neighbor Discovery. Cependant cette solution s'est révélée insuffisante car il est nécessaire de faire un choix entre vitesse de détection et consommation énergétique. C'est pourquoi nous avons proposé d'utiliser le protocole Mobinet pour détecter le mouvement. Les nouveaux résultats montrent que cette combinaison avec Mobile IPv6 permet une détection plus dynamique de la mobilité.

Contrôle de congestion

La sur-écoute liée au médium radio peut également être utilisée pour éviter les congestions. Notre dernière proposition, le protocole multi-couche CLOMAC, fonctionne en combinaison avec un protocole MAC. CLOMAC va utiliser la sur-écoute pour permettre à des nœuds présents dans le voisinage de relayer un paquet à la place du destinataire de niveau 2. Par ce mécanisme opportuniste, notre protocole est capable de construire dynamiquement des chemins alternatifs vers le destinataire final d'un paquet. Pour éviter la retransmission multiple d'un paquet, nous avons fait en sorte que seul le destinataire de niveau 2 puisse le relayer. Cette condition permet d'éviter l'utilisation de mécanisme supplémentaire synchronisant tous les nœuds opportunistes avoisinants. Le changement de destinataire s'effectue en acquittant le préambule du paquet de données.

Nous avons combiné CLOMAC avec le protocole B-MAC et évalué les performances de notre solution par simulation. Les résultats ont montré que, grâce aux nouveaux chemins opportunistes, cette combinaison permet de réduire les congestions en délestant la dorsale de routage. Cette réduction se traduit par une réduction de la perte de paquets, d'une limitation de leur

duplication et surtout par une réduction de la consommation énergétique globale du réseau.

PERSPECTIVES DE RECHERCHE

Pour que l'évaluation de Mobinet combiné à Mobile IPv6 soit complète, il est nécessaire d'en évaluer son impact énergétique. Pour cela, il serait donc intéressant d'utiliser un protocole MAC contenant un mécanisme d'économie d'énergie, tels que X-MAC ou IEEE 802.15.4 en mode synchronisé. Le protocole Mobinet utilise trois valeurs (durée de vie d'un voisin, seuil de changement et intervalle de détection) fixées manuellement dans sa version combinée avec Mobile IPv6. Nous pensons qu'il est possible d'optimiser la détection de mouvements en modifiant dynamiquement ces valeurs. Par exemple, il est possible d'utiliser les données issues d'un capteur de vitesse et de faire varier le seuil de changement. Il serait également possible de faire appel à des mécanismes d'apprentissage simples pour adapter la durée de vie des entrées de la table de voisinage.

A la lumière des résultats obtenus par CLOMAC, il serait intéressant de l'évaluer associé à d'autres protocoles MAC à préambule, tel que X-MAC. Nous pourrions ainsi observer son comportement dans le cas où tout le voisinage n'est pas actif lors de la transmission d'un paquet de données. Des investigations futures pourraient également se focaliser sur la définition d'heuristiques augmentant le nombre de nœuds opportunistes tout en s'assurant que CLOMAC ne génère aucune boucle de routage. Durant nos simulations, nous avons défini le seuil de congestion (CT) à une valeur fixe. En se basant sur des informations provenant du protocole de routage, CLOMAC pourrait faire varier cette valeur en fonction du rôle du nœud dans le réseau. Ainsi un nœud de la dorsale de routage aurait un CT plus faible qu'un nœud feuille. Finalement, il serait intéressant de bénéficier de la plate-forme SensLAB pour évaluer les performances de CLOMAC dans un environnement radio réaliste.

ANNEXES

LISTE DES ACRONYMES

6LBR 6LoWPAN Border Router

6LR 6LoWPAN Router

6LOWPAN IPv6 over Low-Power Wireless Personal Area Networks

ACK acquittement

B-MAC Berkeley MAC

BPSK Binary Phase Shift Keying

CA Collision Avoidance

CBR Constant Bit Rate

CCA Clear Channel Assessment

CLOMAC Cross-Layer Opportunistic MAC

CT Congestion Threshold

CTS Clear To Send

CSMA Carrier Sense Multiple Access

DCF Distributed Coordination Function

ETX Expected Transmission Count

EUI-64 64-bits Extended Unique Identifier

FFD Full Function Device

GTS Guaranted Time Slot

ICMPV6 Internet Control Message Protocol version 6

IEEE Institute of Electrical and Electronics Engineers

IETF Internet Engineering Task Force

IFS Inter-Frame Space

IP Internet Protocol

IPV4 Internet Protocol version 4

IPV6 Internet Protocol version 6

LIFS Long Inter-Frame Space

LQI Link Quality Indication

MAC Medium Access Control

NAT Network Address Translation

OSI Open System Interconnection
PAN Personal Area Network
PRR Packet Receive Rate
RFC Request For Comments
RFD Reduced Function Device
RI-MAC Receiver-Initiated MAC
ROLL Routing Over Low power and Lossy networks
RPL IPv6 Routing Protocol for Low-Power and Lossy Networks
RSSI Radio Strength Signal Indication
RTS Ready To Send
RTT Round-Trip Time
RX Réception
SIFS Small Inter-Frame Space
SLLA source link layer address
TDMA Time Division Multiple Access
TMCP Tree-based Multi-Channel Protocol
TTL Time To Live
TX Transmission
UDP User Datagram Protocol

LISTE DES FIGURES ET TABLES

TABLE DES FIGURES

LISTE DES TABLEAUX

PUBLICATIONS

CONFÉRENCES INTERNATIONALES

Performance Evaluation of Mobile Iv6 Over 6LoWPAN. Damien ROTH, Julien MONTAVONT, Thomas NOEL Dans *PE-WASUN '12 : 9th ACM International Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks*, pages 21–25., Octobre 2012.

Overhearing for Congestion Avoidance in Wireless Sensor Networks. Damien ROTH, Julien MONTAVONT, Thomas NOEL Dans *AdHoc NOW '12 : 10th International Conference on Ad Hoc Networks and Wireless*, pages 1–12., Juillet 2011.

MOBINET : Mobility Management Across Different Wireless Sensor Networks. Damien ROTH, Julien MONTAVONT, Thomas NOEL Dans *WCNC '11 : Wireless Communications and Networking Conference*, pages 1–6. IEEE, Mars 2011.

CASINO : Creating Alea with a Sensor-based Interactive Network (Demo abstract). Julien Beaudaux, Antoine Gallais, Romain Kuntz, Julien Montavont, Thomas Noël, Damien Roth, Fabrice Theoleyre and Erkan Valentin. Dans *Sensys '10 : Conference on Embedded Networked Sensor Systems*. ACM, Novembre 2010.

CONFÉRENCES NATIONALES

MOBINET : gestion de la mobilité à travers différents réseaux de capteurs sans fil. Damien ROTH, Julien MONTAVONT, Thomas NOEL Dans *Algotel '10 : 12èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications*, Juin 2010.

BIBLIOGRAPHIE

[1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *"Computer Networks"*, vol. 52, no. 12, pp. 2292–2330, 2008.

[2] C. A. Boano, J. Brown, Z. He, U. Roedig, and T. Voigt, "Low-Power Radio Communication in Industrial Outdoor Deployments : The Impact of Weather Conditions and ATEX-Compliance," in *Sensor Applications, Experimentation, and Logistics* (N. Komninos, O. Akan, P. Bellavista, J. Cao, F. Dressler, D. Ferrari, M. Gerla, H. Kobayashi, S. Palazzo, S. Sahni, X. S. Shen, M. Stan, J. Xiaohua, A. Zomaya, and G. Coulson, eds.), vol. 29 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 159–176, Springer Berlin Heidelberg, 2010.

[3] G. Tolle, J. Polastre, R. Szewczyk, D. Culler, N. Turner, K. Tu, S. Burgess, T. Dawson, P. Buonadonna, D. Gay, and W. Hong, "A macroscope in the redwoods," in *Proceedings of the 3rd international conference on Embedded networked sensor systems (SenSys'05)*, (New York, NY, USA), pp. 51–63, ACM, Novembre 2005.

[4] I. Vasilescu, K. Kotay, D. Rus, M. Dunbabin, and P. Corke, "Data collection, storage, and retrieval with an underwater sensor network," in *Proceedings of the 3rd international conference on Embedded networked sensor systems (SenSys'05)*, (New York, NY, USA), pp. 154–165, ACM, Novembre 2005.

[5] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Welsh, "Deploying a wireless sensor network on an active volcano," *IEEE Internet Computing*, vol. 10, pp. 18–25, Mars - Avril 2006.

[6] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking : design tradeoffs and early experiences with

ZebraNet," in *Proceedings of the 10th international conference on Architectural support for programming languages and operating systems (ASPLOS-X'02)*, (New York, NY, USA), pp. 96–107, ACM, Octobre 2002.

[7] F. Ingelrest, G. Barrenetxea, G. Schaefer, M. Vetterli, O. Couach, and M. Parlange, "SensorScope : Application-Specific Sensor Network for Environmental Monitoring," *ACM Transactions on Sensor Networks*, vol. 6, pp. 17 :1–17 :32, Février 2010.

[8] J.-H. Huang, S. Amjad, and S. Mishra, "CenWits : a sensor-based loosely coupled search and rescue system using witnesses," in *Proceedings of the 3rd international conference on Embedded networked sensor systems (Sensys'05)*, (New York, NY, USA), pp. 180–191, ACM, Novembre 2005.

[9] M. Rahimi, R. Baer, O. I. Iroezi, J. C. Garcia, J. Warrior, D. Estrin, and M. Srivastava, "Cyclops : in situ image sensing and interpretation in wireless sensor networks," in *Proceedings of the 3rd international conference on Embedded networked sensor systems (SenSys'05)*, (New York, NY, USA), pp. 192–204, ACM, Novembre 2005.

[10] G. Simon, M. Maróti, A. Lédeczi, G. Balogh, B. Kusy, A. Nádas, G. Pap, J. Sallai, and K. Frampton, "Sensor network-based countersniper system," in *Proceedings of the 2nd international conference on Embedded networked sensor systems (SenSys'04)*, (New York, NY, USA), pp. 1–12, ACM, Novembre 2004.

[11] M. Ceriotti, L. Mottola, G. Picco, A. Murphy, S. Guna, M. Corrà, M. Pozzi, D. Zonta, and P. Zanon, "Monitoring Heritage Buildings with Wireless Sensor Networks : The Torre Aquila Deployment," in *Proceedings of the 8th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN'09)*, (Washington, DC, USA), pp. 277–288, IEEE Computer Society, Avril 2009.

[12] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, "Health monitoring of civil infrastructures using wireless sensor networks," in *Proceedings of the 6th international conference on Information processing in sensor networks (IPSN'07)*, (New York, NY, USA), pp. 254–263, ACM, 2007.

[13] T. W. Hnat, V. Srinivasan, J. Lu, T. I. Sookoor, R. Dawson, J. Stankovic, and K. Whitehouse, "The hitchhiker's guide to successful residential sensing deployments," in *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems (SenSys'11)*, (New York, NY, USA), pp. 232–245, ACM, Novembre 2011.

[14] I. Johnstone, J. Nicholson, B. Shehzad, and J. Slipp, "Experiences from a wireless sensor network deployment in a petroleum environment," in *Proceedings of the 2007 international conference on Wireless communications and mobile computing (IWCMC'07)*, (New York, NY, USA), pp. 382–387, ACM, 2007.

[15] L. Krishnamurthy, R. Adler, P. Buonadonna, J. Chhabra, M. Flanigan, N. Kushalnagar, L. Nachman, and M. Yarvis, "Design and deployment of industrial sensor networks : experiences from a semiconductor plant and the north sea," in *Proceedings of the 3rd international conference on Embedded networked sensor systems (SenSys'05)*, (New York, NY, USA), pp. 64–75, ACM, Novembre 2005.

[16] "Coronis Systems." http://www.coronis.com/.

[17] R. Dilmaghani, H. Bobarshad, M. Ghavami, S. Choobkar, and C. Wolfe, "Wireless Sensor Networks for Monitoring Physiological Signals of Multiple Patients," *IEEE Transactions on Biomedical Circuits and Systems (BCS'11)*, vol. 5, pp. 347–356, Août 2011.

[18] J. Ko, J. H. Lim, Y. Chen, R. Musvaloiu-E, A. Terzis, G. M. Masson, T. Gao, W. Destler, L. Selavo, and R. P. Dutton, "MEDiSN : Medical emergency detection in sensor networks," *ACM Transactions on Embedded Computing Systems (TECS'10)*, vol. 10, pp. 11 :1–11 :29, Août 2010.

[19] A. Wood, J. Stankovic, G. Virone, L. Selavo, Z. He, Q. Cao, T. Doan, Y. Wu, L. Fang, and R. Stoleru, "Context-aware wireless sensor networks for assisted living and residential monitoring," *IEEE Network*, vol. 22, pp. 26–33, Juillet - Août 2008.

[20] K. Dantu, M. Rahimi, H. Shah, S. Babel, A. Dhariwal, and G. S. Sukhatme, "Robomote : enabling mobility in sensor networks," in *Proceedings of the 4th international symposium*

on Information processing in sensor networks (IPSN'05), (Piscataway, New Jersey, États-Unis), IEEE Press, 2005.

[21] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things : A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[22] ISO/IEC, "Reference Model of Open Systems Interconnection ; Part 1, Basic Reference Model (incorporating connectionless-mode transmission)," tech. rep., British Standards Institution, 2 Park Street, London W1A 2BS, UK, 1988. BS 6568 : Part 1 : 1988 (≡ ISO 7498–1984 *including Amendment 1*).

[23] IEEE Computer Society, "IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems- Local and Metropolitan Area Networks- Specific Requirements Part 15.4 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)," *IEEE Std 802.15.4-2006 (Revision of IEEE Std 802.15.4-2003)*, pp. 1–305, Septembre 2006.

[24] H. Pham and S. Jha, "An adaptive mobility-aware MAC protocol for sensor networks (MS-MAC)," in *IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MAHSS'04)*, pp. 558–560, Octobre 2004.

[25] R. Kuntz, J. Montavont, and T. Noël, "Improving the medium access in highly mobile Wireless Sensor Networks," *Telecommunication Systems*, pp. 1–22, 2011.

[26] H. Frey, S. Rührup, and I. Stojmenović, "Routing in Wireless Sensor Networks," in *Guide to Wireless Sensor Networks* (A. J. Sammes, ed.), Computer Communications and Networks, pp. 81–111, Springer London, 2009.

[27] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks." IETF Request for Comments (RFC) 4944, Septembre 2007.

[28] J. Hui and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks." IETF Request for Comments (RFC) 6282, Septembre 2011.

[29] C. Perkins, D. Johnson, and J. Arkko, "Mobility Support in IPv6." IETF Request for Comments (RFC) 6275, Juillet 2011.

[30] R. Koodli, "Fast Handovers for Mobile IPv6." RFC 4068 (Experimental), Juillet 2005.

[31] S. Gundavelli, K. Leung, V. Devarapalli, K. Chowdhury, and B. Patil, "Proxy Mobile IPv6." IETF Request for Comments (RFC) 5213, Août 2008.

[32] A. Bachir, M. Dohler, T. Watteyne, and K. Leung, "MAC Essentials for Wireless Sensor Networks," *IEEE Communications Surveys Tutorials*, vol. 12, no. 2, pp. 222 –248, 2010.

[33] "Institute of Electrical and Electronics Engineers." http://www.ieee.org/.

[34] H. Wu and Y. Pan, *Medium Access Control in Wireless Networks*. Nova Science Publishers, 2008.

[35] G. Terrasson, *Contribution à la conception d'un émetteur-récepteur pour microcapteurs autonomes*. PhD thesis, Université de Bordeaux 1, 2008.

[36] Texas Instruments, "Chipcon CC2420 2.4GHz IEEE 802.15.4 RF transceiver." http://www.ti.com/lit/gpn/cc2420, 2007.

[37] Texas Instruments, "CC1101 Low-Power Sub-1GHz RF Transceiver." http://www.ti.com/lit/gpn/cc1101, 2011.

[38] A. R. University, A. Rahman, and P. Gburzynski, "Hidden problems with the hidden node problem," in *Proceedings of the 23th Biennial Symposium on Communications*, pp. 270–273, Mai 2006.

[39] V. Rajendran, K. Obraczka, and J. J. Garcia-Luna-Aceves, "Energy-Efficient, collision-free medium access control for wireless sensor networks," *Wireless Networks*, vol. 12, pp. 63–78, Février 2006.

[40] K. S. J. Pister and L. Doherty, "TSMP : Time synchronized mesh protocol," in *Proceedings of International Symposium on Distributed Sensor Networks (DSN'08)*, pp. 391—398, IEEE/IFIP, Juin 2008.

[41] W. Ye, J. Heidemann, and D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 12, pp. 493–506, Juin 2004.

[42] T. van Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 1st international conference on Embedded networked sensor systems (SenSys'03)*, (New York, NY, USA), pp. 171–180, ACM, Novembre 2003.

[43] C. Cano, B. Bellalta, A. Sfairopoulou, and M. Oliver, "Low energy operation in WSNs : A survey of preamble sampling MAC protocols," *Computer Networks*, vol. 55, no. 15, pp. 3351–3363, 2011.

[44] J. Polastre, J. Hill, and D. Culler, "Versatile Low Power Media Access for Wireless Sensor Networks," in *Proceedings of the 2nd international conference on Embedded networked sensor systems (SenSys'04)*, pp. 95–107, ACM, Novembre 2004.

[45] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC : a short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proceedings of the 4th international conference on Embedded networked sensor systems (SenSys'06)*, (New York, NY, USA), pp. 307–320, ACM, Novembre 2006.

[46] D. Moss and P. Levis, "BoX-MACs : Exploiting Physical and Link Layer Boundaries in Low-Power Networking," Tech. Rep. SING-08-00, Stanford Information Networks Group Technical Report, 2008.

[47] A. El-Hoiydi and J.-D. Decotignie, "WiseMAC : An Ultra Low Power MAC Protocol for Multi-hop Wireless Sensor Networks," in *Algorithmic Aspects of Wireless Sensor Networks* (S. Nikoletseas and J. Rolim, eds.), vol. 3121 of *Lecture Notes in Computer Science*, pp. 18–31, Springer Berlin / Heidelberg, 2004.

[48] M. Zimmerling, F. Ferrari, L. Mottola, T. Voigt, and L. Thiele, "pTunes : runtime parameter adaptation for low-power MAC protocols," in *Proceedings of the 11th international conference on Information Processing in Sensor*

Networks (IPSN'12), (New York, NY, USA), pp. 173–184, ACM, Avril 2012.

[49] R. Kuntz, A. Gallais, and T. Noël, "From versatility to auto-adaptation of the medium access control in wireless sensor networks," *Journal of Parallel and Distributed Computing*, vol. 71, no. 9, pp. 1236–1248, 2011. Special Issue on Advancement of Research in Wireless Access and Mobile Systems.

[50] R. Musaloiu-E., C.-J. M. Liang, and A. Terzis, "Koala : Ultra-Low Power Data Retrieval in Wireless Sensor Networks," in *Proceedings of the 7th international conference on Information processing in sensor networks (IPSN'08)*, (Washington, DC, USA), pp. 421–432, IEEE Computer Society, Avril 2008.

[51] Y. Sun, O. Gurewitz, and D. B. Johnson, "RI-MAC : a receiver-initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks," in *Proceedings of the 6th ACM conference on Embedded network sensor systems (SenSys'08)*, (New York, NY, USA), pp. 1–14, ACM, Novembre 2008.

[52] S. Farahani, *ZigBee Wireless Networks and Transceivers*. Newton, MA, USA : Newnes, 2008.

[53] IEEE Computer Society, "IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999)*, pp. 1–1184, Juin 2007.

[54] M. Ali, T. Suleman, and Z. Uzmi, "MMAC : a mobility-adaptive, collision-free MAC protocol for wireless sensor networks," in *Proceeding of the 24th IEEE International Performance, Computing, and Communications Conference (IPCCC'05)*, pp. 401–407, Avril 2005.

[55] B. Sundararaman, U. Buy, and A. D. Kshemkalyani, "Clock synchronization for wireless sensor networks : a survey," *Elsevier Ad Hoc Networks*, vol. 3, pp. 281–323, Mai 2005.

[56] K. Srinivasan and P. Levis, "RSSI is Under Appreciated," in *Proceedings of the 3rd IEEE Workshop on Embedded Networks*, 2006.

[57] K. Zen, D. Habibi, A. Rassau, and I. Ahmad, "Performance evaluation of IEEE 802.15.4 for mobile sensor networks," in *Proceeding of the 5th IFIP International Conference on Wireless and Optical Communications Networks (WOCN'08)*, pp. 1–5, Mai 2008.

[58] "Internet Engineering Task Force." http://www.ietf.org/.

[59] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification." RFC 2460 (Draft Standard), Décembre 1998.

[60] J. Postel, "Internet Protocol." RFC 791 (Standard), Septembre 1981.

[61] M. Holdrege and P. Srisuresh, "Protocol Complications with the IP Network Address Translator." RFC 3027 (Informational), Janvier 2001.

[62] IEEE Computer Society, "IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems- Local and Metropolitan Area Networks - Specific requirements Part 15.5 : Mesh Topology Capability in Wireless Personal Area Networks (WPANs)," *IEEE Std 802.15.5-2009*, pp. 1–166, May 2009.

[63] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and R. Alexander, "RPL : IPv6 Routing Protocol for Low-Power and Lossy Networks." RFC 6550 (Proposed Standard), Mars 2012.

[64] T. Narten, E. Nordmark, W. Simpson, and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)." IETF Request for Comments (RFC) 4861, Septembre 2007.

[65] Z. Shelby, S. Chakrabarti, and E. Nordmark, "Neighbor Discovery Optimization for Low Power and Lossy Networks (6LoWPAN)." Work in progress, IETF draft-ietf-6lowpan-nd-18, Octobre 2011.

[66] H. Soliman, C. Castelluccia, K. E. Malki, and L. Bellier, "Hierarchical Mobile IPv6 Mobility Management (HMIPv6)." RFC 4140 (Experimental), Août 2005.

[67] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert, "Network Mobility (NEMO) Basic Support Protocol." RFC 3963 (Proposed Standard), Janvier 2005.

[68] R. Silva and J. S. Silva, "An Adaptation Model for Mobile IPv6 support in LoWPANs." Work in progress, IETF draft-silva-6lowpan-mipv6-00, Mai 2011.

[69] T. Camilo, P. Pinto, A. Rodrigues, J. S. Silva, and F. Boavida, "Mobility management in IP-based Wireless Sensor Networks," in *Proceedings of the International Symposium on a World of Wireless, Mobile and Multimedia Networks*, Juin 2008.

[70] A. Jara, R. Silva, J. Silva, M. Zamora, and A. Skarmeta, "Mobile IPv6 over Wireless Sensor Networks (6LoWPAN) : Issues and Feasibility," in *Proceedings of the 7th European Conference on Wireless Sensor Networks (EWSN'10)*, Février 2010.

[71] J. H. Kim, R. Haw, and C. S. Hong, "Development of a framework to support network-based mobility of 6LoWPAN sensor device for mobile healthcare system," in *Digest of Technical Papers International Conference on Consumer Electronics (ICCE'10)*, Janvier 2010.

[72] J. Kim, R. Haw, E. Cho, C. Hong, and S. Lee, "A 6LoWPAN Sensor Node Mobility Scheme Based on Proxy Mobile IPv6," *IEEE Transactions on Mobile Computing*, vol. PP, no. 99, p. 1, 2011.

[73] M. Ha, D. Kim, S. H. Kim, and S. Hong, "Inter-MARIO : A Fast and Seamless Mobility Protocol to Support Inter-Pan Handover in 6LoWPAN," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM'10)*, pp. 1–6, Décembre 2010.

[74] A. Jara, M. Zamora, and A. Skarmeta, "HWSN6 : Hospital Wireless Sensor Networks Based on 6LoWPAN Technology : Mobility and Fault Tolerance Management," in *Proceedings of International Conference on Computational Science and Engineering (CSE'09)*, vol. 2, pp. 879–884, Août 2009.

[75] D. Roth, J. Montavont, and T. Noel, "MOBINET : gestion de la mobilité à travers différents réseaux de capteurs sans fil," in *12èmes Rencontres Francophones sur les*

Aspects Algorithmiques de Télécommunications (AlgoTel'10)) (M. G. Potop-Butucaru and H. Rivano, eds.), (Belle Dune, France), Juin 2010.

[76] D. Roth, J. Montavont, and T. Noel, "MOBINET : Mobility management across different wireless sensor networks," in *Proceedings of the Wireless Communications and Networking Conference (WCNC'11)*, pp. 351–356, Mars 2011.

[77] Q. H. Lilia Paradis, "A Survey of Fault Management in Wireless Sensor," *Journal of Network and Systems Management*, vol. 15, pp. 171–190, 2007.

[78] J. Martocci, P. D. Mil, N. Riou, and W. Vermeylen, "Building Automation Routing Requirements in Low-Power and Lossy Networks." RFC 5867 (Informational), Juin 2010.

[79] A. Mei and J. Stefa, "Routing in Outer Space," in *Proceedings of the 27th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'08)*, pp. 2234–2242, Avril 2008.

[80] A. Fraboulet, G. Chelius, and E. Fleury, "Worldsens : Development and Prototyping Tools for Application Specific Wireless Sensors Networks," in *Proceeding of the 6th International Symposium on Information Processing in Sensor Networks (IPSN'07)*, pp. 176–185, Avril 2007.

[81] H. Karl and A. Willig, *Protocols and Architectures for Wireless Sensor Networks*. John Wiley & Sons, 2005.

[82] T. Camp, J. Boleng, and V. Davies, "A Survey of Mobility Models for Ad Hoc Network Research," *Wireless Communications & Mobile Computing (WCMC)*, vol. 2, pp. 483–502, 2002. Special Issue on Mobile AD HOC Networking : Research, Trends and Applications.

[83] J. Polastre, R. Szewczyk, and D. Culler, "Telos : Enabling Ultra-low Power Wireless Research," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN'05)*, pp. 364–369, Avril 2005.

[84] D. Roth, J. Montavont, and T. Noel, "Performance Evaluation of Mobile Iv6 Over 6LoWPAN," in *Proceedings of the 9th ACM workshop on Performance evaluation of wireless ad*

hoc, sensor, and ubiquitous networks (PE-WASUN'12), (New York, NY, USA), ACM, Octobre 2012.

[85] A. Dunkels, B. Grönvall, and T. Voigt, "Contiki - A Lightweight and Flexible Operating System for Tiny Networked Sensors," in *Proceedings of the 29th IEEE Conference on Local Computer Networks (LCN'04)*, pp. 455–462, Novembre 2004.

[86] A. Dunkels, "Full TCP/IP for 8-bit architectures," in *Proceedings of the 1st International Conference on Mobile Systems, Applications and Services (MobiSys'03)*, pp. 85–98, Mai 2003.

[87] A. Dunkels, F. Österlind, N. Tsiftes, and Z. He, "Software-based sensor node energy estimation," in *Proceedings of the 5th international conference on Embedded networked sensor systems (SenSys'07)*, (New York, NY, USA), pp. 409–410, ACM, Novembre 2007.

[88] Texas Instruments, "MSP430 : 16-bit Ultra-Low-Power MCU." http://www.ti.com/product/msp430f1611, 2011.

[89] Crossbow, "TelosB Mote Platform datasheet." http://www.willow.co.uk/TelosB_Datasheet.pdf, 2006.

[90] C. Margi, B. de Oliveira, G. de Sousa, M. Simplicio, P. Barreto, T. Carvalho, M. Nä andslund, and R. Gold, "Impact of Operating Systems on Wireless Sensor Networks (Security) Applications and Testbeds," in *Proceedings of 19th International Conference on Computer Communications and Networks (ICCCN'10)*, pp. 1–6, Août 2010.

[91] D. Roth, J. Montavont, and T. Noël, "Overhearing for Congestion Avoidance in Wireless Sensor Networks," in *Ad-hoc, Mobile, and Wireless Networks* (H. Frey, X. Li, and S. Ruehrup, eds.), vol. 6811 of *Lecture Notes in Computer Science*, pp. 131–144, Springer Berlin / Heidelberg, 2011.

[92] Y. Wu, J. Stankovic, T. He, and S. Lin, "Realistic and Efficient Multi-Channel Communications in Wireless Sensor Networks," in *Proceedings of the 27th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'08)*, pp. 1193–1201, Avril 2008.

[93] C.-Y. Wan, S. B. Eisenman, A. T. Campbell, and J. Crowcroft, "Siphon : overload traffic management using multi-radio virtual sinks in sensor networks," in *Proceedings of the 3rd international conference on Embedded networked sensor systems (Sensys'05)*, (New York, NY, USA), pp. 116–129, ACM, Novembre 2005.

[94] G.-S. Ahn, S. G. Hong, E. Miluzzo, A. T. Campbell, and F. Cuomo, "Funneling-MAC : a localized, sink-oriented MAC for boosting fidelity in sensor networks," in *Proceedings of the 4th international conference on Embedded networked sensor systems (SenSys'06)*, (New York, NY, USA), pp. 293–306, ACM, Octobre 2006.

[95] J. Luo and J.-P. Hubaux, "Joint mobility and routing for lifetime elongation in wireless sensor networks," in *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'05)*, vol. 3, pp. 1735–1746, Mars 2005.

[96] D. Jea, A. Somasundara, and M. Srivastava, "Multiple Controlled Mobile Elements (Data Mules) for Data Collection in Sensor Networks," in *Distributed Computing in Sensor Systems* (V. Prasanna, S. Iyengar, P. Spirakis, and M. Welsh, eds.), vol. 3560 of *Lecture Notes in Computer Science*, pp. 466–466, Springer Berlin / Heidelberg, 2005.

[97] S. Lohs, R. Karnapke, and J. Nolte, "Link Stability in a Wireless Sensor Network – An Experimental Study," in *Sensor Systems and Software* (F. Martins, L. Lopes, H. Paulino, O. Akan, P. Bellavista, J. Cao, F. Dressler, D. Ferrari, M. Gerla, H. Kobayashi, S. Palazzo, S. Sahni, X. S. Shen, M. Stan, J. Xiaohua, A. Zomaya, and G. Coulson, eds.), vol. 102 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 146–161, Springer Berlin Heidelberg, 2012.

[98] S. Biswas and R. Morris, "ExOR : opportunistic multi-hop routing for wireless networks," *SIGCOMM Comput. Commun. Rev.*, vol. 35, pp. 133–144, Août 2005.

[99] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," *Wireless Networks*, vol. 11, pp. 419–434, 2005.

[100] G. Schaefer, F. Ingelrest, and M. Vetterli, "Potentials of Opportunistic Routing in Energy-Constrained Wireless Sensor Networks," in *Proceedings of the 6th European Conference on Wireless Sensor Networks (EWSN'09)*, (Berlin, Heidelberg), pp. 118–133, Springer-Verlag, 2009.

[101] H. Dubois-Ferrière, M. Grossglauser, and M. Vetterli, "Valuable detours : least-cost anypath routing," *IEEE/ACM Transactions on Networking (TON)*, vol. 19, pp. 333–346, Avril 2011.

[102] S. Liu, M. Sha, and L. Huang, "ORAS : Opportunistic routing with asynchronous sleep in Wireless Sensor Networks," in *Proceedings of the 2nd International Conference on Future Computer and Communication (ICFCC'10)*, vol. 3, pp. V3–234–V3–238, Mai 2010.

[103] M. Autenrieth and H. Frey, "PaderMAC : A Low-Power, Low-Latency MAC Layer with Opportunistic Forwarding Support for Wireless Sensor Networks," in *Ad-hoc, Mobile, and Wireless Networks* (H. Frey, X. Li, and S. Ruehrup, eds.), vol. 6811 of *Lecture Notes in Computer Science*, pp. 117–130, Springer Berlin / Heidelberg, 2011.

[104] O. Landsiedel, E. Ghadimi, S. Duquennoy, and M. Johansson, "Low power, low delay : opportunistic routing meets duty cycling," in *Proceedings of the 11th international conference on Information Processing in Sensor Networks (IPSN'12)*, (New York, NY, USA), pp. 185–196, ACM, Avril 2012.

[105] B. Pavković, F. Theoleyre, and A. Duda, "Multipath opportunistic RPL routing over IEEE 802.15.4," in *Proceedings of the 14th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems (MSWiM'11)*, (New York, NY, USA), pp. 179–186, ACM, Juillet 2011.

[106] S. Unterschütz, C. Renner, and V. Turau, "Opportunistic, Receiver-Initiated Data-Collection Protocol," in *Wireless Sensor Networks* (G. Picco and W. Heinzelman, eds.), vol. 7158 of *Lecture Notes in Computer Science*, pp. 1–16, Springer Berlin / Heidelberg, 2012.

[107] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, "Collection tree protocol," in *Proceedings of the 7th*

ACM Conference on Embedded Networked Sensor Systems (SenSys'09), (New York, NY, USA), pp. 1–14, ACM, Novembre 2009.