



TRAVAIL D'ETUDE ET DE RECHERCHE

M1 SCIENCES ET INGÉNIERIE DES RÉSEAUX, DE L'INTERNET ET DES  
SYSTÈMES

CMI INFORMATIQUE, SYSTÈMES ET RÉSEAUX

---

# Support de la multi-domiciliation dans l'Internet des Objets

---

*Auteur*

Paul HENG

*Encadrant*

Julien MONTAVONT

PRINTEMPS 2020

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Présentation du contexte</b>	<b>2</b>
<b>3</b>	<b>Présentation du problème</b>	<b>2</b>
<b>4</b>	<b>Pile protocolaire standardisée pour l’IoT</b>	<b>4</b>
4.1	6LoWPAN . . . . .	4
4.2	Le protocole de routage : RPL . . . . .	5
4.2.1	Généralités . . . . .	5
4.2.2	Messages de contrôle . . . . .	6
4.2.3	Création d’un DODAG . . . . .	7
4.3	Conclusion de l’étude protocolaire . . . . .	8
<b>5</b>	<b>La multi-domiciliation d’un LLN dans la littérature</b>	<b>8</b>
5.1	Syn-RPL . . . . .	9
5.1.1	Principe . . . . .	9
5.1.2	Amorçage du protocole . . . . .	9
5.1.3	Gestion du basculement . . . . .	10
5.1.4	Observations . . . . .	11
5.2	RPL-NPT-LB . . . . .	12
5.2.1	Découverte . . . . .	12
5.2.2	Load Balancing . . . . .	12
5.2.3	Network Prefix Translation . . . . .	13
5.2.4	Observations . . . . .	13
5.3	Conclusion de l’état de l’art . . . . .	14
<b>6</b>	<b>Expérimentations avec RPL-NPT-LB</b>	<b>14</b>
6.1	Résultats comparés à ceux de la littérature . . . . .	15
6.2	Essais avec plus de noeuds . . . . .	17
6.3	Changement de déclencheur du mode congestionné . . . . .	19
6.4	Améliorer le mécanisme de sélection des noeuds à rediriger . . . . .	22
6.5	Bilan des travaux expérimentaux . . . . .	22
6.6	Perspectives . . . . .	23
<b>7</b>	<b>Conclusion</b>	<b>23</b>

## 1 Introduction

Le Travail d'Etude et de Recherche est une unité d'enseignement visant à introduire les étudiants au monde de la recherche à travers une étude et un travail sur un sujet d'actualité lié au milieu. J'ai choisi le sujet du support de la multi-domiciliation dans l'Internet des Objets (IoT) afin de pouvoir découvrir des notions que j'ignore encore. L'IoT est une notion dont on entend souvent parler récemment, mais sans savoir exactement à quoi cela correspond. Ce travail était donc pour moi l'occasion de me pencher sur le sujet afin de mieux comprendre ce qui est défini comme l'IoT, quelles sont ses applications et ses contraintes.

## 2 Présentation du contexte

Les objets connectés, comme des capteurs par exemple, se caractérisent par des capacités limitées, que ce soit en terme de puissance de calcul, de mémoire ou d'énergie. Cela s'explique principalement de par leur taille et coût réduits. Ces objets peuvent être inter-connectés entre eux, formant ainsi ce qui est appelé dans la littérature un Low-power and Lossy Network (LLN). L'Internet des Objets (IoT), vient du fait qu'il est possible de relier un LLN à l'Internet via une passerelle, communément appelée routeur de bordure (Border Router, ou BR). Pour faire face aux contraintes liées à ce type de réseaux, une pile de protocoles différents de la pile TCP/IP traditionnelle est utilisée, avec des standards dédiés aux LLN proposés par l'IEEE et l'IETF. En effet, les contraintes telles que la mémoire disponible et la consommation énergétique rendent l'utilisation de protocoles comme TCP (durée variable d'un RTT, maintien de session) peu adaptée.

Une topologie traditionnelle de LLN est multi-sauts, certains noeuds jouant de rôle de noeuds intermédiaires afin d'acheminer le trafic jusqu'au BR. Cela forme un graphe acyclique orienté vers une destination (Destination Oriented Directed Acyclic Graph, ou DODAG). L'utilisation de noeuds intermédiaire est nécessaire de par la limite de puissance d'émission des objets (noeuds), mais cela permet également de réduire l'empreinte énergétique des communications, car plus un noeud est proche, moins la consommation d'énergie pour communiquer est importante.

## 3 Présentation du problème

Un réseau contraint (LLN) peut être relié à l'Internet via un routeur de bordure (BR), formant ainsi ce que l'on appelle l'Internet des Objets. La figure 1 illustre une topologie dans l'Internet des Objets, le BR étant la racine du DODAG.

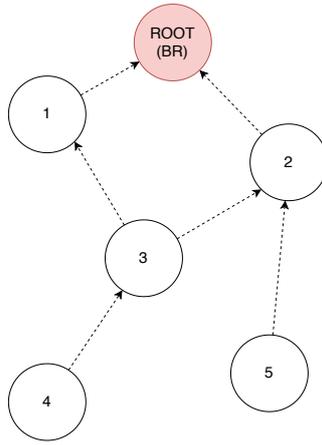


FIGURE 1 : DODAG dans un LLN

Toutefois, la présence d'un unique BR dans le réseau introduit un unique point de défaillance. En effet, si ce BR vient à tomber en panne, c'est l'entièreté du réseau qui se retrouve coupée d'Internet.

Par ailleurs, la topologie multi-sauts des LLN introduit un autre problème : la structure en DODAG implique que chaque noeud doit faire suivre le trafic de son sous-DODAG vers le BR. Ainsi, les noeuds les plus proches du BR disposent d'un sous-DODAG plus grand, avec plus d'enfants et donc de trafic à faire suivre. Cela se traduit par des émissions plus régulières, donc une consommation d'énergie plus élevée, et ils sont donc susceptibles de se décharger plus rapidement, déconnectant ainsi tous leur sous-DODAG d'Internet. Ce phénomène est appelé l'effet entonnoir (Funnel Effect) et est illustré sur la figure 2. De plus, un tel phénomène est susceptible d'augmenter la latence, ce qui est problématique pour des applications critiques (dans le domaine de la santé par exemple).

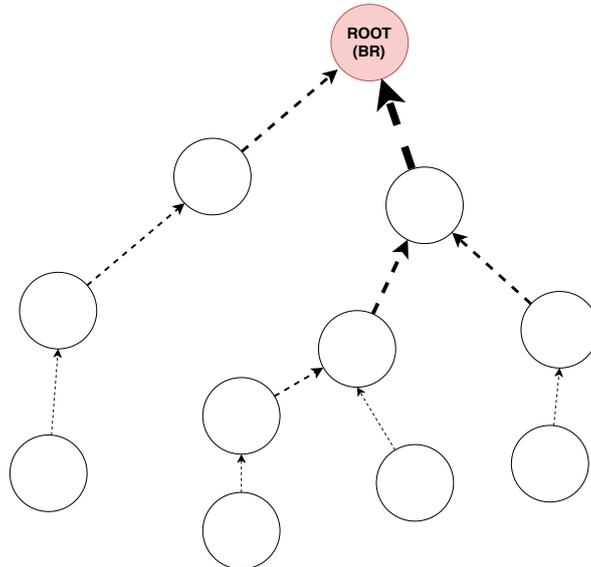


FIGURE 2 : Effet entonnoir (plus la flèche est épaisse, plus il y a de trafic)

Une solution pour pallier à ces problèmes est la multi-domiciliation du LLN.

La multi-domiciliation d'un réseau consiste à introduire multiplier le nombre de passerelles

vers Internet dans ce réseau, ces dernières pouvant avoir des caractéristiques différentes (réseaux d’opérateurs, réseau cellulaire). Cette redondance permet d’obtenir de multiples bénéfices. D’une part, avec plusieurs passerelles de sortie, il est possible de partager voire d’équilibrer la charge dans le réseau, et de ce fait, d’augmenter le débit. De plus, la présence de plusieurs passerelles permet de conserver une connectivité même si l’une d’elle vient à tomber en panne. Multi-domicilier un réseau contribue donc à le rendre plus robuste à la charge et aux aléas.

Ajouter un support de la multi-domiciliation dans les LLN permettrait donc de résoudre ces deux problèmes majeurs. Toutefois, dans la pile standardisée, il n’y a pas de spécification précise concernant la présence de plusieurs BR dans le réseau.

Dans la section 4 seront présentés les protocoles standardisés pour l’IoT ainsi que leur fonctionnement. La section 5 présente des solutions existantes dans la littérature permettant d’introduire la multi-domiciliation dans un LLN. Enfin, la section 6 sera consacrée à l’étude expérimentale d’une de ces solutions.

## 4 Pile protocolaire standardisée pour l’IoT

Pour tenir compte des contraintes matérielles des appareils évoluant dans un LLN, une pile protocolaire spécifique à été standardisée. La figure 3 montre les différents protocoles de cette dernière. Le standard 802.15.4 de l’IEEE [4] détaille les couches physique et MAC. Le partage du médium peut-être assuré par différents algorithmes comme CSMA/CA slotté ou TSCH, et des périodes de communications (duty cycle) et de repos être définies, permettant aux noeuds d’éteindre leur radio pendant une certaine période afin de réduire leur consommation énergétique. Ici, nous nous intéresserons plus particulièrement à la couche IP en étudiant le protocole de routage, RPL, mais également à une couche d’adaptation avant ce dernier afin de pouvoir associer les pré-requis d’IPv6 et de 802.15.4.

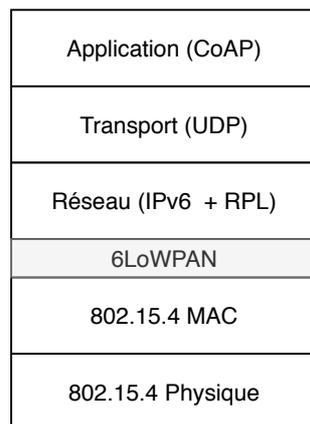


FIGURE 3 : Pile standardisée pour l’IoT

### 4.1 6LoWPAN

Les trames 802.15.4 sont limitées en terme de taille à 127 octets, ce qui pose problème notamment avec IPv6, dont le MTU minimum spécifié est de 1280 octets. De plus, en tenant compte de

la taille des en-têtes 802.15.4 et des éventuels mécanismes de sécurité, les données sont potentiellement limitées à 41 octets par trame, couche transport incluse. Pour transmettre des paquets IPv6 sur des trames 802.15.4, il est donc nécessaire de recourir à des mécanismes comme la compression et la fragmentation. Pour cela, l'IETF spécifie une couche d'adaptation entre les couches liaison et réseau, appelée 6LoWPAN [17] [14], comme illustré sur la figure 3. Ce standard propose via des en-têtes dans les paquets, des mécanismes de fragmentation avant la couche IP et de compression.

La compression, appelée LOWPAN\_IPHC [14], permet de réduire de l'en-tête IPv6. Pour cela, certains champs disposent de valeurs par défaut (hop-limit par exemple), d'autres sont réduits (compression du préfixe lien-local sur 1 bit), et certains sont tout simplement omis comme la version d'IP par exemple, étant donné que 6LoWPAN est conçu pour IPv6. LOWPAN\_IPHC est une amélioration de la spécification de compression proposée originalement, LOWPAN\_HC1 [17], qui était plus limitée en ne gérant au niveau de l'adressage que les adresses IPv6 lien-local par exemple. La figure 4 montre un exemple de compression de paquet.

6LoWPAN spécifie également une en-tête mesh, permettant de faire du routage au niveau 2. On parle alors de routage mesh-under, en opposition au routage classique effectué au niveau 3 appelé route-over.

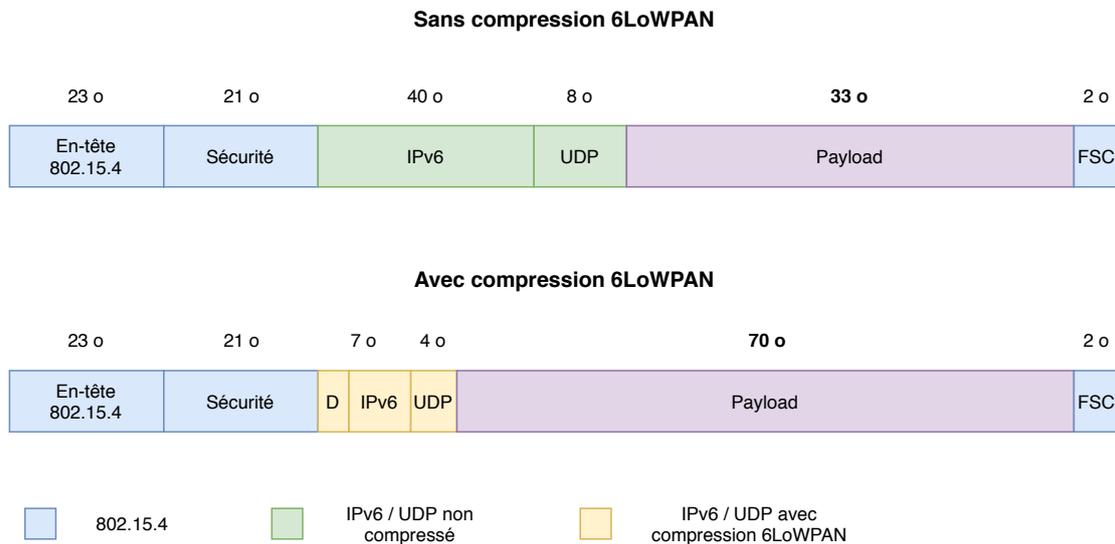


FIGURE 4 : Compression IPHC

6LoWPAN est donc un mécanisme extrêmement utile afin de pouvoir envoyer des paquets IPv6 sur des trames 802.15.4. La compression étant effectuée avant la couche IP, cela permet de compresser non seulement les données, mais également une partie de l'en-tête IPv6 pour réduire au maximum la taille du paquet à envoyer.

## 4.2 Le protocole de routage : RPL

### 4.2.1 Généralités

La pile standardisée pour l'IoT utilise pour la couche 3 le protocole RPL (IPv6 Routing Protocol for Low-Power and Lossy Networks) [11].

Il s'agit d'un protocole de routage de type vecteur de distance, car la puissance de calcul

et la mémoire nécessaires pour le bon fonctionnement d'un protocole à états de liens sont trop importantes pour les noeuds qui composent un LLN.

RPL construit une topologie sous la forme d'un DODAG (cf. figure 1) et supporte 3 types de communications :

- Multipoint-à-Point (plusieurs appareils vers un point central), qui est le mode par défaut et toujours supporté
- Point-à-Multipoint (point central vers plusieurs appareils)
- Point-à-Point (communication entre appareils dans le réseau)

La hiérarchie utilisée par le protocole est organisée sous forme de rangs (la racine ayant un rang égal à 0), qui sont calculés à l'aide d'une fonction objectif (OF). L'OF est une fonction (partagée au sein d'une instance de RPL) permettant de calculer le rang d'un noeud et de sélectionner son parent préféré, en se basant sur certaines métriques ou contraintes [15]. Ces métriques peuvent être des métriques de liens (débit, latence, Expected Transmission Count (ETX), nombre de sauts), ou des métriques de noeuds (charge CPU du noeud, sur batterie ou non, batterie restante). Une métrique « couleur » peut également être définie manuellement. Certaines OF sont standardisées par l'IETF, comme l'Objective Function Zero (OF0) [20] ou Minimum Rank with Hysteresis Objective Function (MRHOF) [12].

RPL se découpe en instances, chacune ayant son propre identifiant (RPLInstanceID). Une instance peut être composée de plusieurs DODAG, chacun ayant également un identifiant unique (DODAGID, correspondant à l'identifiant de la racine) mais pouvant avoir plusieurs versions (DODAGVersionNumber) selon le rattachement des noeuds comme illustré sur la figure 5.

Un DODAG est donc identifié de manière unique par le triplet (RPLInstanceID, DODAGID, DODAGVersionNumber).

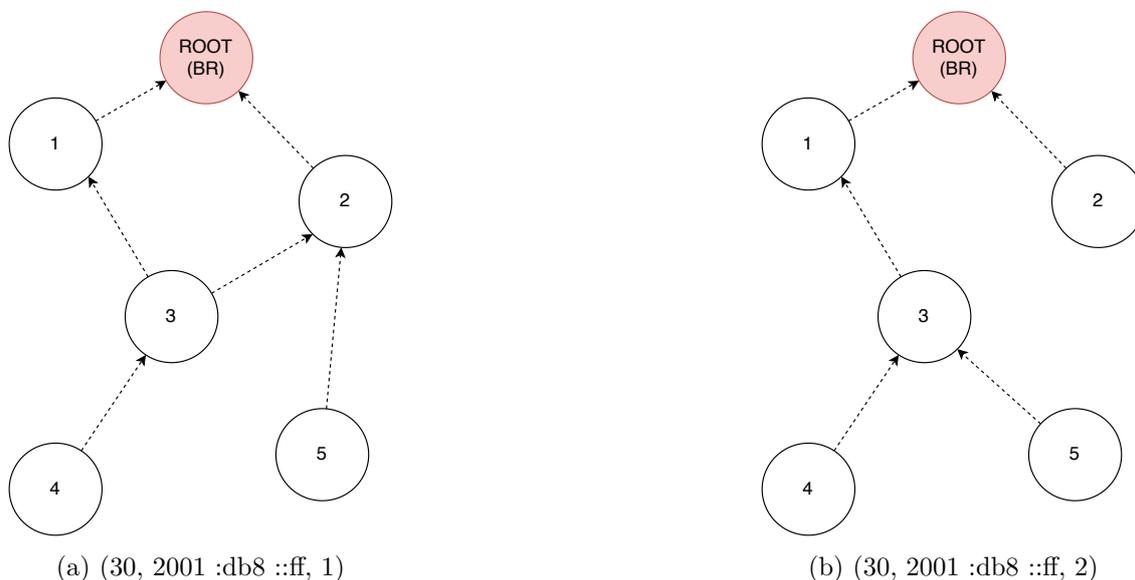


FIGURE 5 : Différentes versions d'un DODAG avec le triplet identifiant associé

#### 4.2.2 Messages de contrôle

Les messages de contrôle RPL sont des messages ICMPv6, et ils sont au nombre de 3 :

- DODAG Information Object (DIO) : contient des informations générales sur l'instance RPL (triplet identifiant, rang de l'émetteur), et précise l'OF utilisée. Le DIO peut contenir une option DAG Metric Container, utilisée pour l'envoi de métriques (utilisées par l'OF et/ou pour faire de la qualité de service). Les DIO sont notamment utilisés pour la génération des routes ascendantes (des noeuds vers la racine).
- DODAG Information Sollicitation (DIS) : utilisé pour demander explicitement à un noeud d'envoyer un DIO.
- Destination Advertisement Object (DAO) : généré par les noeuds (hors-racine), qui contient les informations sur le noeud qui l'a généré. Les DAO sont utilisés en particulier pour la génération des routes descendantes (racine vers les noeuds).

### 4.2.3 Création d'un DODAG

La création d'un DODAG commence par la création des routes ascendantes. Pour cela, des DIO sont envoyés en multicast, dont le premier émetteur est la racine. A la réception d'un DIO, un noeud (i) ajoute l'émetteur à sa liste de parents, (ii) calcule son propre rang à l'aide de l'OF spécifiée dans le DIO, (iii) élit son parent préféré, et (iv) retransmet à son tour le DIO en remplaçant le champ rang du message précédemment reçu par le sien.

Les routes descendantes sont créées à partir des DAO générés par les noeuds, qui sont ensuite relayés jusqu'à la racine. RPL définit deux modes opératoires pour la gestion des routes descendantes :

- Mode non-stockant : dans ce mode, seule la racine dispose d'une vue d'ensemble du réseau et connaît les routes vers les différents noeuds. Dans ce mode, les noeuds qui reçoivent des DAO se contentent de le relayer jusqu'à racine en ajoutant simplement ses propres adresses dans le message. Les communications point-à-point sont donc effectués via un routage par source dans ce mode.
- Mode stockant : dans ce mode, les noeuds intermédiaires conservent les informations contenues dans les DAO qu'ils reçoivent, et maintiennent une table de routage qui leur est propre. Cela apporte donc de meilleures performances pour les communications point-à-point, au détriment d'une consommation d'énergie et de mémoire supplémentaire.

La Figure 6 illustre la différence entre les deux modes. En mode non-stockant, le message envoyé par D à destination de E remonte d'abord à la source (racine du DODAG), pour ensuite être renvoyé vers E. En mode stockant, B dispose d'une table de routage et connaît donc une route vers E, le message ne remonte donc que jusqu'à B pour ensuite être renvoyé en direction de E.

Le maintien du DODAG s'effectue via l'envoi périodique de DIO pour signaler que les noeuds sont toujours actifs. RPL utilise un trickle timer [16] pour l'envoi des DIO afin d'optimiser la fréquence de transmission des messages selon la stabilité du réseau, et ainsi limiter les envois quand ce n'est pas nécessaire pour réduire l'énergie consommée. Un intervalle d'émission minimal à respecter est défini, et ce dernier augmente jusqu'à un certain seuil tant que le réseau est stable (pas de noeuds qui tombent en panne par exemple). Lorsqu'une fluctuation est détectée, l'intervalle reprend sa valeur minimale.

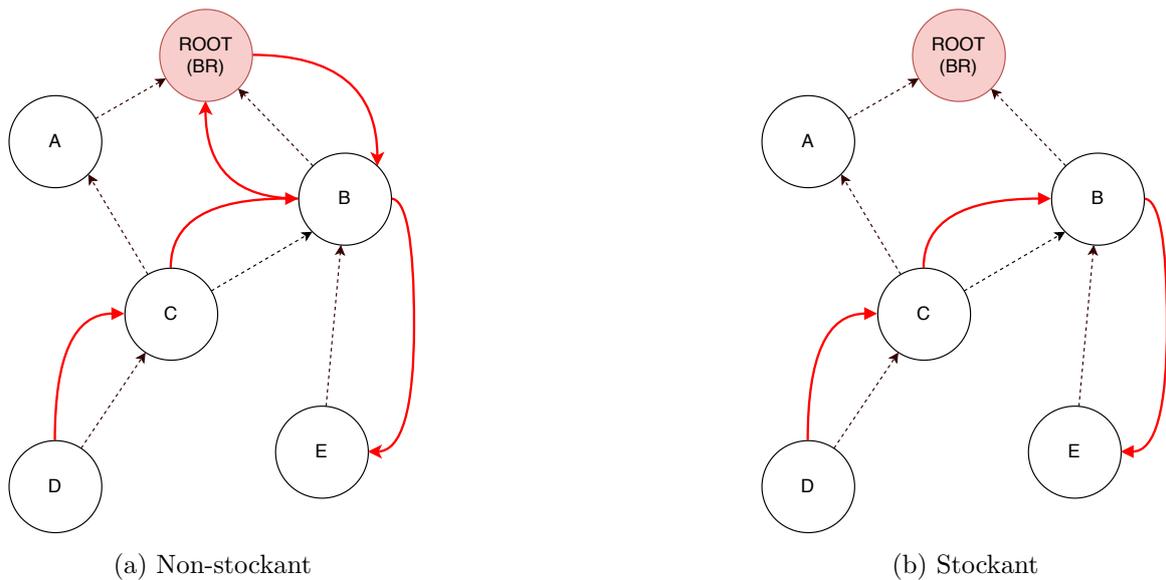


FIGURE 6 : Mode non-stockant vs. stockant

### 4.3 Conclusion de l'étude protocolaire

RPL supporte la présence de plusieurs routeurs de bordure, que ce soit à travers plusieurs instances du protocole, ou plusieurs DODAG au sein d'une même instance. Toutefois, aucune spécification n'est précisée concernant la présence de multiples BRs. De même, le protocole dispose de mécanismes de récupération quand un noeud classique tombe en panne, mais ces derniers ne s'appliquent pas si la défaillance provient du BR. Les solutions existantes dans la littérature spécifiant un mécanisme de multi-domiciliation dans un LLN avec plusieurs BRs sont étudiées dans la section suivante.

## 5 La multi-domiciliation d'un LLN dans la littérature

Afin de répondre à la problématique évoquée dans 3, les protocoles présentés dans la littérature doivent répondre à certains critères. Ces protocoles doivent d'une part spécifier un mécanisme de basculement, c'est-à-dire un mécanisme permettant aux noeuds de changer leur BR par défaut s'il vient à tomber en panne pour conserver leur connectivité. D'autre part, pour réduire l'effet entonnoir, ils doivent également proposer un moyen de partager ou d'équilibrer la charge de chaque BR, et donc indirectement des noeuds voisins.

En plus de simplement ajouter un mécanisme permettant aux noeuds de basculer sur un autre BR, il est intéressant de tenir compte du fait que les différents BR ne sont pas forcément rattachés au même réseau d'opérateur et utilisent donc des préfixes IPv6 différents. Il faut donc (i) que les noeuds conservent un préfixe cohérent lors du changement de BR afin d'éviter l'Ingress Filtering, et (ii) rediriger les paquets entrants destinés à l'ancien BR vers le nouveau.

Plusieurs solutions permettant de multiplier de nombre de BR existent dans la littérature.

Dans [6], une entité externe, le registrar, joue le rôle de noeud virtuel (VN) comme spécifié dans [22]. Ce dernier opère comme racine du DODAG et fait le lien entre plusieurs instances RPL. Le premier BR se signalant auprès du registrar initialise son DODAG et envoie ses paramètres à ce

dernier. Les BR voulant joindre la racine virtuelle par la suite reçoivent la configuration du DODAG depuis le registrar avant de commencer d'émission des DIOs. Cette solution permet le partage de charge entre les BR via les mécanismes inhérents à RPL, étant donné que le DODAG est commun (sélection du meilleur parent via le rang). Toutefois, cette solution ne prend pas en compte la possibilité que les préfixes gérés par les BR soient différents, et ne spécifie aucun comportement à adopter en cas de panne d'un des BR.

D'autres solutions présentées dans [9] et [8] sont également basées sur RPL et permettent l'introduction de plusieurs BRs dans un LLN, avec des mécanismes permettant aux noeuds de changer dynamiquement le sous-DODAG auquel ils sont rattachés. Les résultats montrent notamment une amélioration de la qualité du réseau grâce au partage de charge, mais ici encore, aucune de ces solutions ne spécifie de comportement à adopter en cas de panne d'un BR.

Une autre solution proposée dans [7] consiste à relier plusieurs BRs à un réseau Ethernet commun, et d'ajouter un appareil hors du LLN jouant le rôle de racine du DODAG dans ce réseau. Les BRs agissent alors comme de simples noeuds, leur panne pouvant alors être gérée par les mécanismes de récupération prévus par RPL. Cette solution ne prend toutefois pas non plus en compte la gestion de préfixes différents.

Enfin, il existe des travaux présentant des solutions tenant compte de l'ensemble des contraintes (panne d'un BR, préfixes différents et partage de charge), qui sont détaillées dans les sections suivantes.

## 5.1 Syn-RPL

### 5.1.1 Principe

Syn-RPL [19] est une solution utilisant un noeud virtuel tout comme dans [6] afin de créer un DODAG commun entre les différents BRs, qui sont alors ancrés au VN comme des noeuds classiques. Comme les BRs deviennent des noeuds classiques, les mécanismes internes à RPL peuvent s'appliquer, à savoir la sélection du BR préféré pour chaque noeud via un calcul de chemins avec la fonction objectif et les métriques utilisées. Ainsi, le partage de charge s'effectue automatiquement et permet de limiter l'effet entonnoir. De même, les mécanismes de récupérations en cas de panne d'un noeud spécifiés par RPL sont applicables dans ce cas de figure si un BR tombe en panne.

Syn-RPL introduit une nouvelle entité jouant le rôle de noeud virtuel, l'Anchor Agent (AA). Ce dernier dispose de paramètres nécessaires à la configuration du DODAG, qui sont partagés avec les différents BRs. Il sert également de relais pour le trafic, le trafic entrant étant routé vers l'AA dans un premier temps, ce qui permet de maintenir la connectivité même en cas de panne d'un BR. Le trafic sortant du LLN est également routé via l'AA, mais le mode stockant peut être utilisé pour les communications internes. Pour cela, des tunnels IPv6 bi-directionnels entre l'AA et les BRs sont utilisés.

### 5.1.2 Amorçage du protocole

La mise en place de Syn-RPL commence par la pré-configuration de l'AA et des BRs. L'AA est configuré avec la DODAGVersion (cf. section 4.2.1), l'OF et la métrique de routage utilisées, une durée de vie (une fois dépassée, un BR est considéré comme injoignable), les identifiants

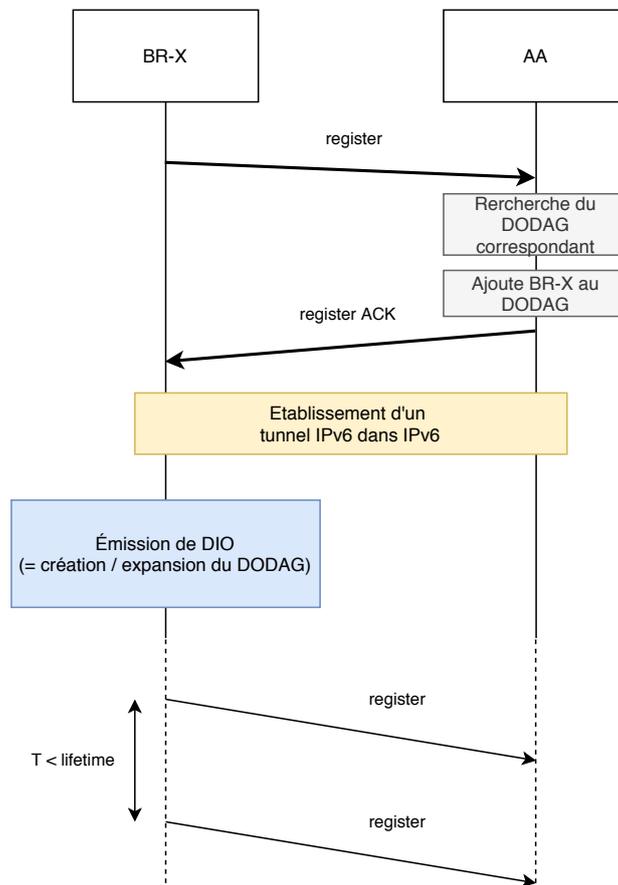


FIGURE 7 : Amorçage de Syn-RPL

des BRs (l'adresse EUI-64 du BR par exemple), les préfixes IPv6 à déléguer aux BR et enfin un secret partagé afin d'authentifier les BRs souhaitant se rattacher à l'AA. Les BRs sont également configurés avec leur identifiant respectif, l'adresse IPv6 de l'AA et le secret partagé.

La figure 7 détaille la phase d'amorçage. Dans un premier temps, les BRs souhaitant se rattacher à l'AA envoient un message *register* pour ce dernier avec son identifiant. À la réception du message, l'AA vérifie la présence de l'identifiant dans sa base de données et lui répond avec un *register ACK*. Si la demande est valide, ce dernier contient les informations permettant au BR de construire le DODAG. À partir de ce moment, la création ou l'expansion du DODAG peut commencer, le BR envoie des DIO et reçoit des DAO après rattachement des noeuds au DODAG. Le tunnel IPv6 entre le BR et l'AA est également créé. Des *register* sont ensuite envoyés périodiquement à l'AA pour signaler que le BR est toujours joignable (période inférieure à la durée de vie configurée sur l'AA).

### 5.1.3 Gestion du basculement

Le mécanisme de basculement est décrit sur la figure 8. Quand un BR est détecté comme injoignable, les noeuds qui y sont rattachés envoient des DIO à leurs enfants avec un rang infini (route poisoning) pour les avertir que le BR n'est plus disponible. A la réception de ces DIO, les noeuds peuvent accepter les DIO des sous-DODAG voisins pour s'y rattacher, puis se signalent avec des DAO. Les noeuds nouvellement rattachés ne changent pas leur adresse IPv6, mais le BR

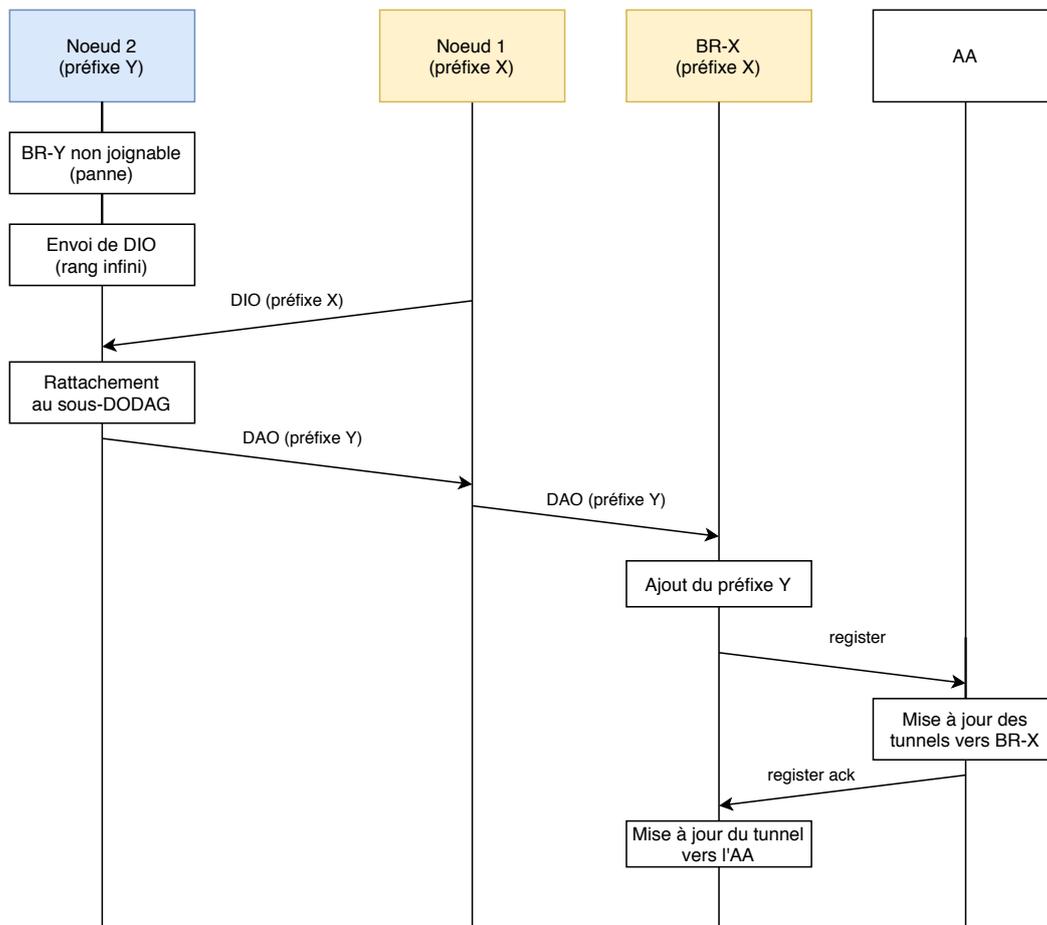


FIGURE 8 : Basculement dans Syn-RPL

recevant des DAO avec des adresses d'autres préfixes demande à l'AA via des register la délégation de ces préfixes. Enfin, l'AA met à jour ses tunnels qui étaient liés au BR en panne pour les relier au nouveau BR. A la réception d'un register ACK, le BR ayant initié la demande peut mettre à jour son propre tunnel pour gérer les nouveaux préfixes.

#### 5.1.4 Observations

Syn-RPL propose une solution afin d'introduire la multi-domiciliation dans un LLN, mais déplace l'unique point de défaillance, qui n'est alors plus le BR mais l'AA, et ne le supprime pas. Toutefois, l'AA n'étant pas soumis aux mêmes contraintes que les BR (il peut être connecté à un réseau IPv6 plus traditionnel, en connexion filaire), des solutions déjà existantes, comme Virtual Router Redundancy Protocol (VRRP) [18], permettent d'introduire de la redondance sur ce type d'appareils. De plus, Syn-RPL présente l'avantage de ne nécessiter des changements que sur les BRs (en plus d'introduire l'AA), ce qui rend la solution facilement déployable sur un réseau d'objets déjà existant.

La question de la suppression totale de l'unique point de défaillance est traitée avec la solution présentée dans la section suivante.

## 5.2 RPL-NPT-LB

Une autre solution existante dans la littérature est RPL-NPT-LB [10]. Cette solution n'introduit pas de nouvelle entité dans le réseau, mais propose un système de coopération entre les différents BRs. Cette approche « distribuée » permet la suppression de l'unique point de défaillance, ce qui n'était pas le cas avec une solution comme Syn-RPL qui reste centralisée (au niveau de l'Anchor Agent). Pour tenir compte des différents préfixes gérés par les routeurs, RPL-NPT-LB dispose d'une adaptation de Network Prefix Translation (NPT) [21], et dispose également d'un mécanisme d'équilibrage de charge. Le protocole repose également sur le maintien d'un DODAG commun, mais cette fois en utilisant plusieurs instances RPL, ce qui permet de savoir à quel BR est rattaché un noeud (un BR par instance).

### 5.2.1 Découverte

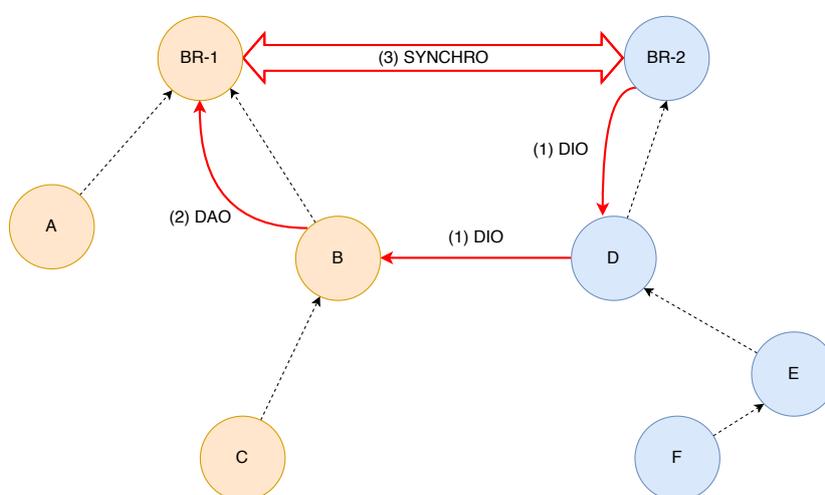


FIGURE 9 : Processus de découverte RPL-NPT-LB (1 couleur / instance)

Le mécanisme de découverte du réseau de RPL-NPT-LB utilise la sur-écoute (overhearing) des DIO reçus d'un DODAG voisin. Lorsqu'un noeud reçoit un DIO appartenant à une autre instance RPL (1), il transforme ce DIO en DAO, en y ajoutant des informations relatives à l'instance voisine comme l'adresse IP du BR voisin et le transmet à son propre BR. A la réception du DAO, les BR vont synchroniser leurs informations afin de maintenir une vue globale et commune du réseau. Ce processus est illustré sur la figure 9.

### 5.2.2 Load Balancing

La figure 10 décrit le fonctionnement de l'équilibrage de charge de RPL-NPT-LB, qui se base sur le mode non-stockant de RPL (routage par source). Lors du processus de découverte, les noeuds recevant (1) et transformant (2) les DIO voisins en DAO se signalent comme noeuds dits « redirigeables » à leur BR. Lors de la synchronisation entre les différents BRs, ces derniers échangent, en plus de l'ensemble des destinations existantes dans le DODAG commun, un ensemble de noeuds redirigeables.

RPL-NPT-LB introduit un nouveau mode de fonctionnement pour les BR, le mode congestionné, qui peut être activé selon certains critères comme la taille du sous-DODAG ou le taux

de pertes dans la file d'attente. Quand un BR entre dans ce mode, il sélectionne un ou plusieurs noeuds parmi son ensemble de noeuds redirigeables, et leur envoie un nouveau message, le DODAG Redirection Solicitation (DRS) (3). Quand un noeud reçoit un DRS, il envoie un DIS (4) à un noeud d'un DODAG voisin afin de mettre à jour ses informations (car elles peuvent être obsolètes), puis se rattache au DODAG voisin à la réception du DIO (5 - 6).

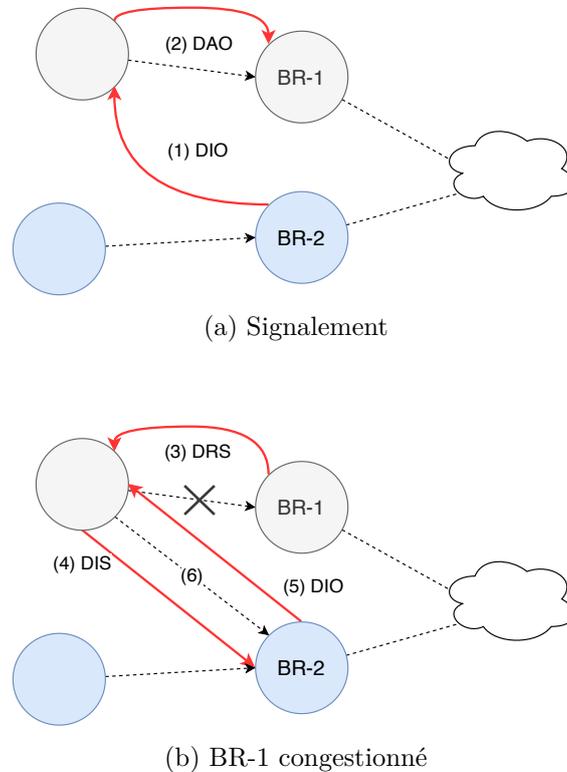


FIGURE 10 : Load Balancing dans RPL-NPT-LB

### 5.2.3 Network Prefix Translation

Pour répondre à cette problématique des différents réseaux d'accès, RPL-NPT-LB utilise une adaptation de Network Prefix Translation [21]. Le principe consiste à utiliser un préfixe local [13] dans le DODAG commun, les différents BRs effectuant la translation entre les adresses locales au sein du LLN et les adresses globales avec les préfixes qu'ils gèrent lors d'une communication avec Internet (local vers global pour le trafic sortant, et inversement). Étant donné que les noeuds sont tous dans le même réseau, ils n'ont pas besoin de changer de préfixe quand leur BR tombe en panne, les BRs en coopération opérant comme routes de secours pour les préfixes de leurs voisins. La figure 11 illustre la translation entre le préfixe local et les préfixes globaux.

### 5.2.4 Observations

Les résultats expérimentaux d'une implémentation de RPL-NPT-LB [10] se traduisent par une augmentation de la bande passante globale du réseau, mais également par une réduction du taux d'erreurs, impliquant ainsi moins de retransmissions et donc une consommation énergétique réduite sur la majorité des noeuds. Toutefois, un désavantage de cette solution par rapport à Syn-RPL est la nécessité de modifier tous les noeuds, ce qui peut être problématique sur des réseaux d'objets

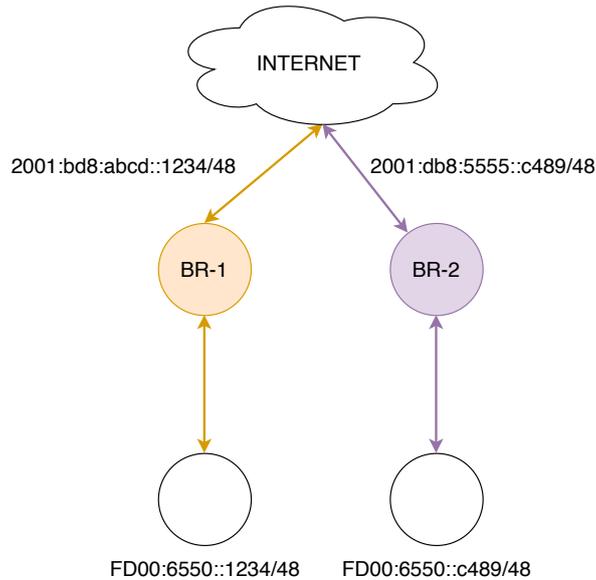


FIGURE 11 : Network Prefix Translation

déjà installés. Par ailleurs, la synchronisation nécessaire entre les différents BRs augmente leur consommation en ressources, afin de maintenir la vue globale du réseau.

### 5.3 Conclusion de l'état de l'art

Nous avons donc vu que plusieurs protocoles ont été proposés pour permettre la redondance de BRs dans un LLN. Cependant, les solutions existantes ne spécifient pas toujours de comportement à adopter lors de la panne d'un BR, et très peu d'entre elles tiennent compte de la gestion de préfixes différents. Certaines approches sont centralisées et se reposent sur une entité supplémentaire comme Syn-RPL, qui ne corrigent pas directement le problème de l'unique point de défaillance mais le délèguent à une installation plus simple à redonder, tandis que d'autres proposent une approche distribuée, comme RPL-NPT-LB, permettant de supprimer directement l'unique point de défaillance au sein du LLN, mais ce type de solutions sollicite plus de ressources de la part des BRs.

Intéressés par le mécanisme dynamique de partage de charge de RPL-NPT-LB, ainsi que par son approche se limitant au LLN, nous avons choisi ce protocole pour une phase expérimentale.

## 6 Expérimentations avec RPL-NPT-LB

Pour cette partie, nous avons utilisé les travaux réalisés dans [10].

Ces travaux ont été réalisés sur Contiki [2], qui est un système d'exploitation léger et open-source dédié à l'IoT. Il s'agit d'un des systèmes d'exploitation les plus utilisés pour l'IoT, qui dispose d'un support d'IPv6, de 6LoWPAN, ainsi que de sa propre implémentation de RPL, Contiki-RPL.

Nos travaux ont également été déployés sur une plateforme d'expérimentation à large échelle appelée FIT/IoT-Lab [5]. Les noeuds de cette plateforme sont répartis sur plusieurs sites en France et sont en accès libre. Pour ces expérimentations, ce sont des noeuds M3 dont les caractéristiques techniques sont disponibles sur [3] qui ont été utilisés.

## 6.1 Résultats comparés à ceux de la littérature

Pour nos premiers essais, nous avons repris les paramètres expérimentaux de [10], qui sont détaillés dans la table 1.

MAC	802.15.4 + CSMA/CA avec acquittements
Taille de file d'attente (émission)	1 paquet
Duty cycle	Aucun
Trafic émis	1 paquet UDP (20 octets) / seconde
Puissance d'émission	3 dBm
Seuil de réception	-60 dBm
Mode RPL	Non-stockant
OF	MRHOF (avec ETX)
Déclencheur du mode congestionné	Taille du sous-DODAG
Durée	60 minutes

TABLE 1 : Paramètres expérimentaux de base

Nous avons commencé par redéployer le protocole de test employé dans l'article, avec 10 noeuds, dont deux routeurs de bordure (BR 18 et BR 53). Dans l'expérience, le but étant de tester le comportement lorsque l'on ajoute un BR à un LLN déjà existant, le BR 53 démarre 60 secondes plus tard. Les figures 12a et 12b montrent les différentes topologies obtenues suite à l'expérience. Un arc plus épais indique que la topologie est apparue plus de fois. Nous observons des résultats similaires, sauf pour le noeud 26. Dans [10], il s'agissait toujours d'un noeud feuille. Dans nos expérimentations, il joue également le rôle de noeud intermédiaire. Nous avons effectué une série de 30 expériences pour chaque protocole.

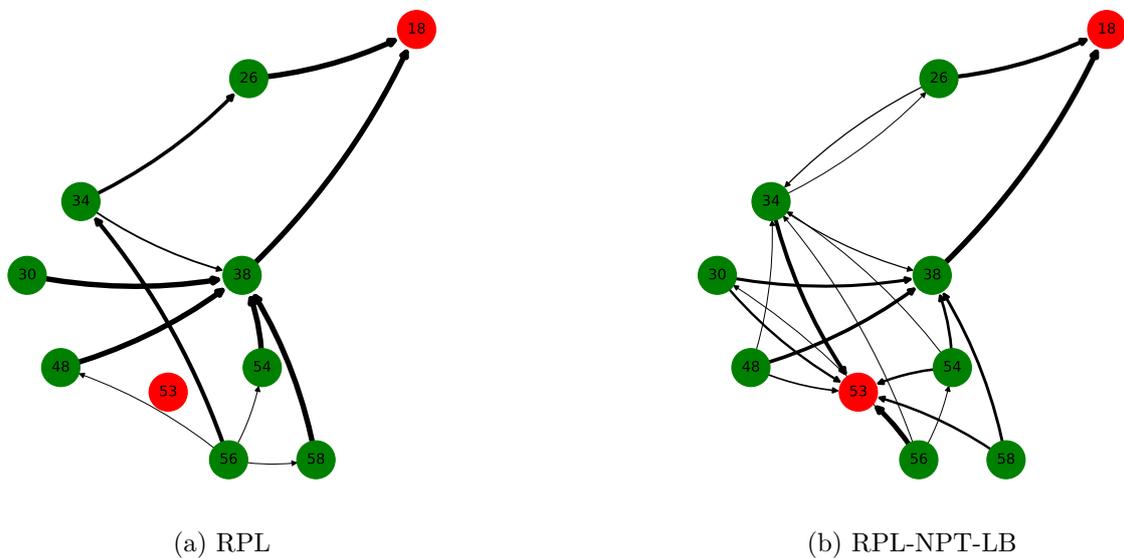


FIGURE 12 : Topologies pour 10 noeuds

La figure 13 détaille le nombre de transmissions MAC moyennes moyennes, à gauche pour RPL, à droite pour RPL-NPT-LB. Nous avons mesuré le nombre de transmissions correctement réalisées, non-acquittées (timeout sur l'acquittement par exemple), d'acquittements transmis ainsi que le nombre de collisions sur la réception ces derniers. Nous avons également mesuré le nombre de pertes

dans la file d'attente de transmission de chaque noeud. Nous constatons bien une diminution du nombre de transmissions globales avec RPL-NPT-LB. Le mécanisme de partage de charge contribue à réduire le nombre de transmissions non-acquittées, ainsi que le nombre de collisions. De fait, nous observons également une diminution du nombre de ré-émissions.

On notera toutefois une différence sur le noeud 26, qui ici jouait le rôle de noeud intermédiaire pour le noeud 34, d'où l'envoi d'acquittements. De plus, il s'est également fait redirigé dans les expériences avec NPT-LB vers ce même noeud.

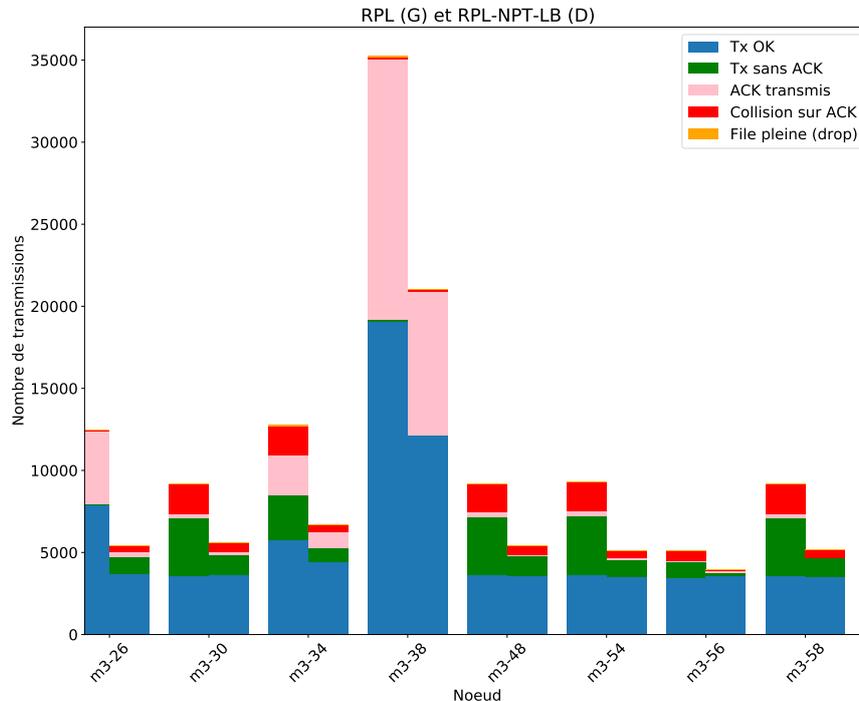


FIGURE 13 : Transmissions MAC pour 10 noeuds

Nous avons ensuite étudié le Packet Delivery Ratio (PDR) saut-à-saut de chaque noeud. Ce dernier correspond au rapport entre le nombre total de paquets envoyés par un noeud à ses voisins directs et le nombre de paquets que ces derniers ont effectivement reçu de ce noeud, soit  $\frac{TotalEnvoi}{TotalReception} * 100$ . La figure 14 montre la différence de PDR entre RPL-NPT-LB et RPL pour chaque noeud.

Nous constatons une amélioration globale du PDR, de 20% à 40% généralement, sauf pour le noeud 38, qui n'a que très peu de concurrence sur le médium pour ses communications avec le BR 18. Nous observons également une diminution du PDR dans certaines expériences pour le noeud 26. En effet, la redirection de ce dernier vers le BR 53 n'étant pas optimale car cela introduit des sauts supplémentaires, alors qu'il est à portée directe du BR 18.

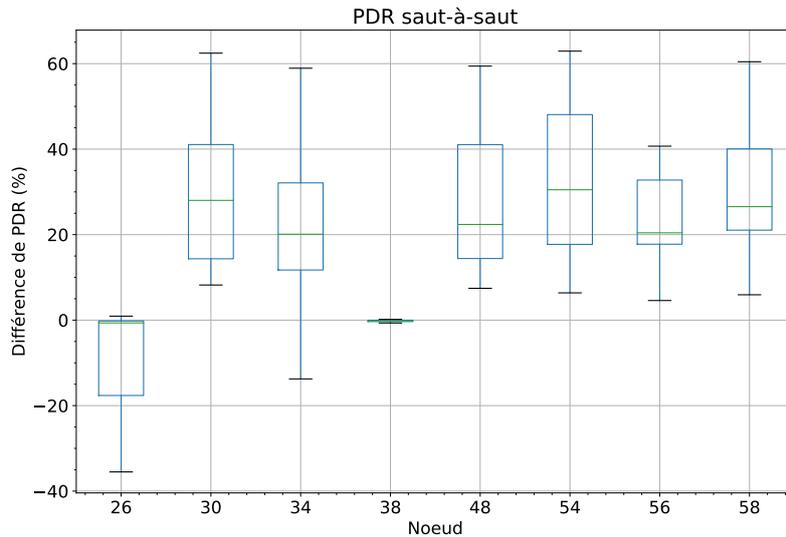


FIGURE 14 : PDR saut-à-saut avec 10 noeuds

Mis à part la différence de comportement du noeud 26, qui est possiblement due à une modification de la position du noeud sur le site d'expérimentation (ou à une erreur de notre part), nos résultats sont cohérents avec ceux exposés dans la littérature.

## 6.2 Essais avec plus de noeuds

Les résultats de base s'étant avérés concluants, nous avons voulu essayer avec des configurations différentes, avec plus de noeuds et plus de BRs pour voir si l'amélioration constatée était toujours présente.

Pour cela 16 noeuds classiques ont été ajoutés sur l'ensemble du site, portant le total à 26 noeuds. Un tel nombre de noeuds donnant lieu à un réseau très congestionné, nous avons testé RPL-NPT-LB avec deux BR, puis avec un troisième (donc 27 noeuds au total) démarrant en même temps que le second. 20 expériences ont été effectuées pour chaque situation.

A nouveau, nous étudions les transmissions MAC, qui sont présentes sur la figure 15, à gauche pour 2 BR et à droite pour 3 BR. Compte-tenu du nombre de noeuds, les erreurs liées aux acquittements sont majoritairement des collisions. Nous observons que l'ajout d'un BR supplémentaire contribue largement à réduire ces dernières, en plus de la réduction du nombre de retransmissions évoqué précédemment. Ici, nous constatons également des pertes dues à la file d'attente pleine, qui sont également réduites avec l'ajout du troisième BR.

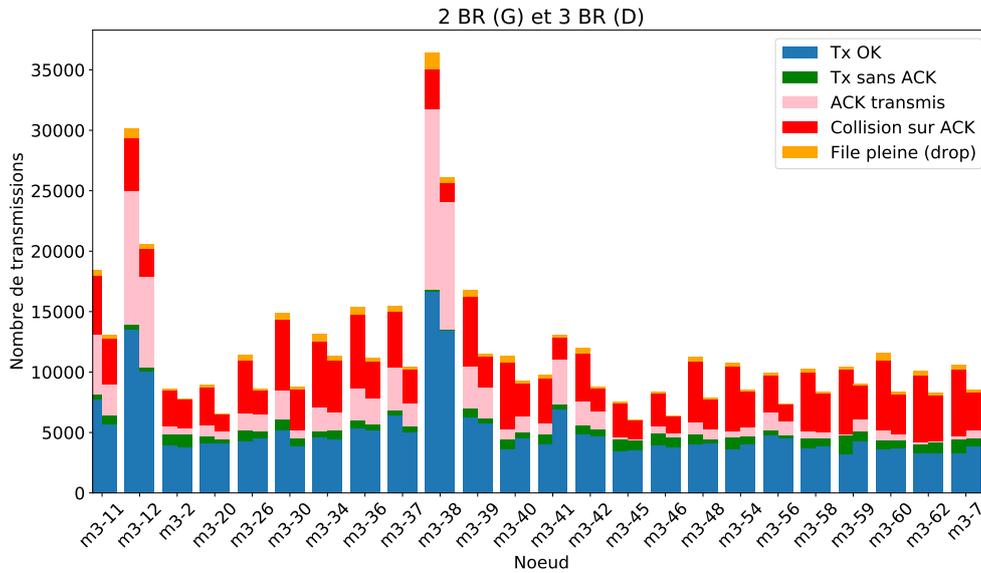


FIGURE 15 : Transmissions MAC pour 27 noeuds avec RPL-NPT-LB

Nous nous sommes également intéressés à l'évolution de la bande passante, qui est illustrée sur la figure 16. Nous avons mesuré la bande passante moyenne à partir de  $t = 1500s$  (afin que le mécanisme de partage de charge ait eu le temps d'agir) jusqu'à la fin de l'expérience, et l'efficacité du partage de charge se traduit par une augmentation du débit de l'ordre de 60 octets par seconde pour l'ajout du BR supplémentaire.

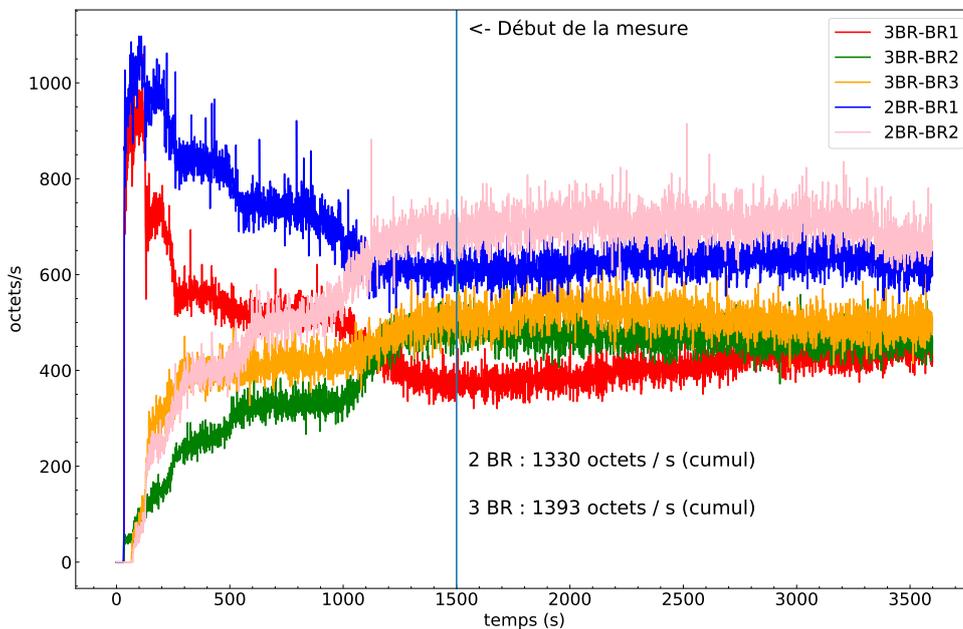


FIGURE 16 : bande passante moyenne pour 27 noeuds avec RPL-NPT-LB

### 6.3 Changement de déclencheur du mode congestionné

Après avoir pu confirmer l’efficacité de RPL-NPT-LB avec plus de noeuds, nous nous sommes intéressés aux limites d’un déclencheur comme la taille du sous-DODAG. En effet, se baser uniquement sur le nombre d’enfants d’un BR ne prend pas en compte la situation de ces derniers, il est possible que certains noeuds enfants soient plus sollicités que d’autres, il serait alors intéressant de rediriger ces derniers en priorité.

Pour illustrer ce comportement, nous sommes repartis de l’expérience de base (cf. 6.1 et avons rajouté 2 noeuds simples (27 et 28) et un BR (noeud 46) supplémentaire. Les noeuds supplémentaires sont proches du BR 18 et sont donc moins soumis à des phénomènes de congestion que les autres. Ici, le BR 46 se réveille après 60 secondes, et le BR 53 se réveille encore 60 secondes après.

Pour tenir compte de la situation des noeuds enfants, nous avons implémenté un déclencheur alternative du mode congestionné d’un BR. Nous avons ajouté une nouvelle information aux DAO envoyés, qui indiquent si le noeud est lui-même considéré comme congestionné. Cela délègue donc la détection de congestion aux noeuds enfants. Un noeud s’annonce comme congestionné à partir d’un seuil de pertes dans sa file d’attente de transmissions entre deux émissions de DAO.

Chaque fois qu’un BR reçoit un DAO, il vérifie dans sa liste de noeuds s’il y a des noeuds congestionnés, et essaye de rediriger la moitié des enfants de ces derniers. Un noeud congestionné est marqué comme traité si certains de ses enfants ont été redirigés, et le BR ne vérifiera plus ce noeud jusqu’au prochain DAO reçu de la part de ce dernier, afin d’éviter les redirections en chaîne.

Nous avons ensuite comparé les deux déclencheurs dans une série de 20 expériences chacun. La configuration des expériences suivantes est disponible dans la table 2. Nous avons utilisé une valeur fixe pour la taille du sous-DODAG afin de simuler un environnement où le BR 18 aurait des noeuds enfants supplémentaires hors de portée d’une autre instance.

Nombre de noeuds	13
Seuil du déclencheur (taille du sous-DODAG)	6 noeuds (BR inclus)
Seuil du déclencheur (pertes dans la file)	10 paquets entre 2 DAO

TABLE 2 : Configuration utilisé pour les expériences sur déclencheur

Les topologies obtenues sont montrées dans les figures 17a pour NPT-LB avec taille du sous-DODAG, et dans 17b avec les pertes dans la file comme déclencheur.

En utilisant la taille, les noeuds 26, 27 et 28 détectent l’instance du BR 46, et sont ainsi redirigés alors qu’ils se sont pas spécialement chargés. Quand le BR 53 se réveille, les noeuds à porté détectent l’instance et 3 autres noeuds sont redirigés. Avec les pertes dans la file comme déclencheur, comme les noeuds 26, 27 et 28 ne sont pas redirigés vers une autre instance étant donné qu’ils ne subissent pas de pertes. En revanche, certains noeuds ( noeud 38 notamment) doivent faire suivre le trafic de plusieurs enfants vers le BR 18, des pertes dans leur file d’attente ont donc lieu et ces derniers se signalent comme congestionnés, entraînant ainsi la redirection de certains de leurs enfants vers le BR 53.

La figure 18 montre les transmissions MAC selon le déclencheur utilisé. On constate une diminution des transmissions générales en utilisant la file d’attente comme déclencheur, notamment en ce qui concerne les transmissions non acquittées et les collisions. En effet, étant donné que ce sont les noeuds à proximité du BR 53 qui sont principalement concernés par la congestion, ce sont eux

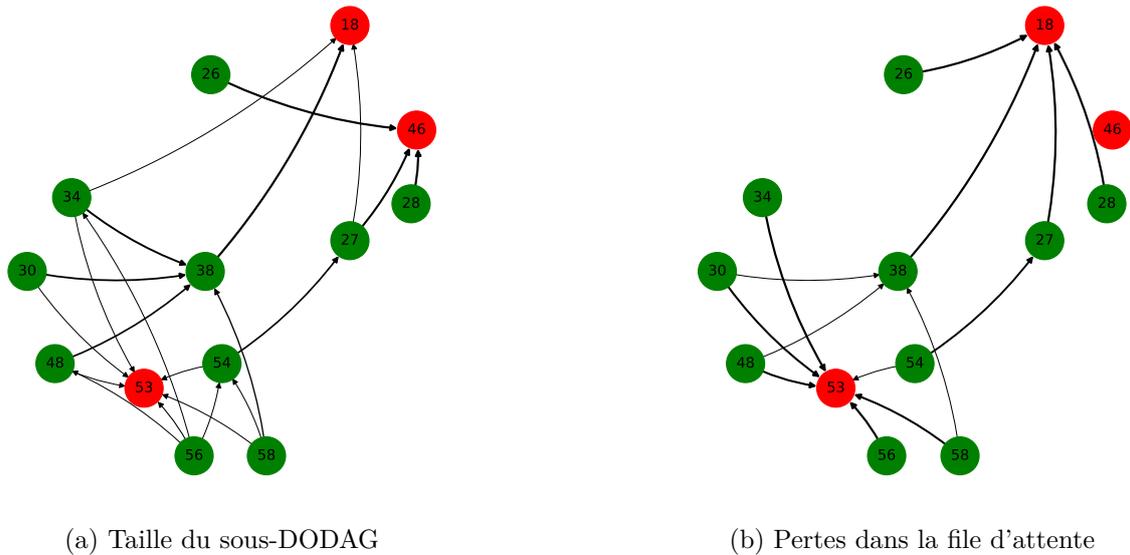


FIGURE 17 : Topologies pour 12 noeuds selon le déclencheur

qui sont redirigés vers ce BR quand il se réveille, alors qu'en utilisant la taille du sous-DODAG comme déclencheur, rediriger d'abord les noeuds 27 et 28 vers le BR 46 ne contribue pas (ou très peu) à réduire le phénomène de congestion.

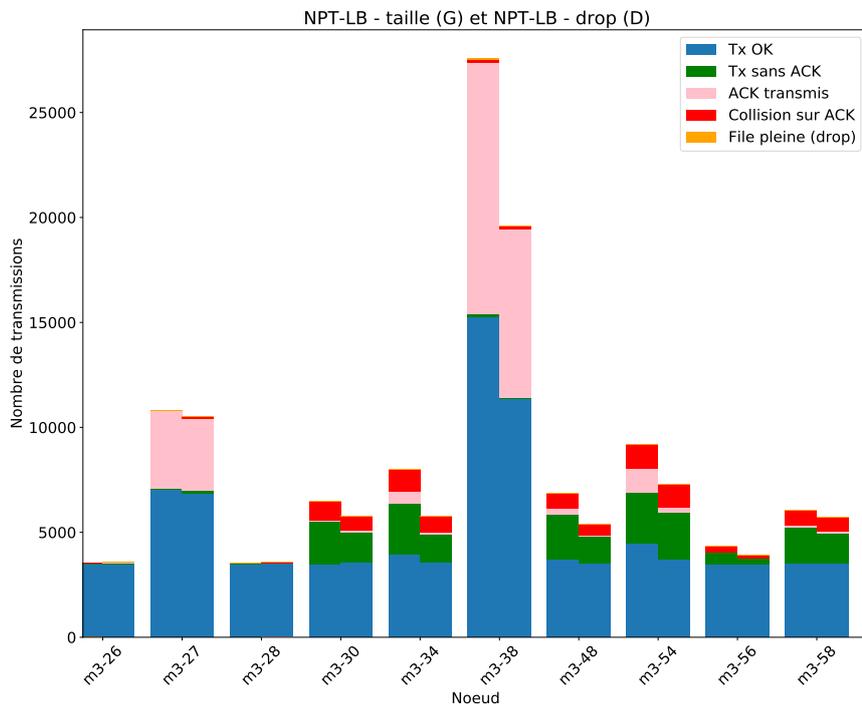


FIGURE 18 : Transmissions MAC pour 12 noeuds selon le déclencheur utilisé

De même, nous avons étudié l'évolution la bande passante moyenne pour chaque série d'expériences, représentée sur la figure 19. Le signalement de la congestion étant évolutif, l'expérience ne

converge pas à partir d'un certain temps ou d'un certain nombre de redirections. De fait, nous nous sommes intéressés à la bande passante moyenne sur la durée totale des expériences et grâce à la redirection de plus de noeuds vers l'instance du BR 53, nous constatons une amélioration légère de 4 octets par seconde.

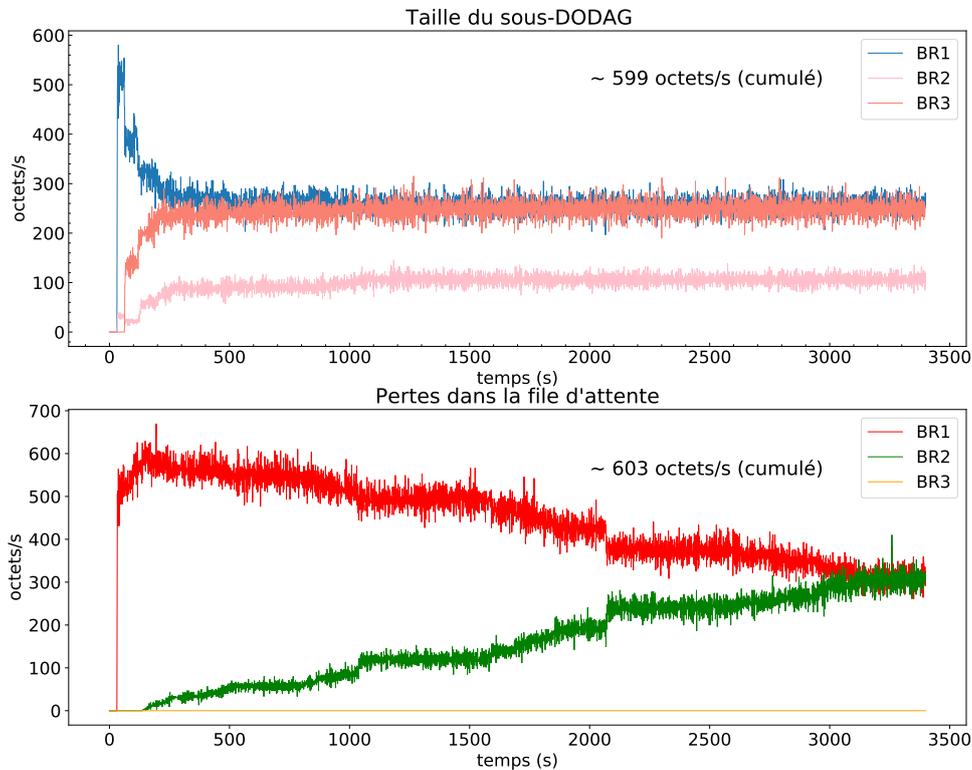


FIGURE 19 : Bande passante moyenne pour 12 noeuds selon le déclencheur

De manière analogue à l'expérience de base, nous avons calculé la différence de PDR saut-à-saut pour chaque noeud entre le déclencheur pertes et le déclencheur taille, visible sur la figure 20. Ici encore, nous constatons une amélioration du PDR global avec le déclencheur pertes. Le médium étant peu occupé pour les communications directes avec les BRs 18 et 46, il n'y a pas de différence pour les noeuds 26, 27, 28 et 38. En revanche, il y a beaucoup de concurrence pour les communications vers le noeud 38, et les redirections supplémentaires dans cette partie du réseau permet une augmentation du PDR pour les autres noeuds.

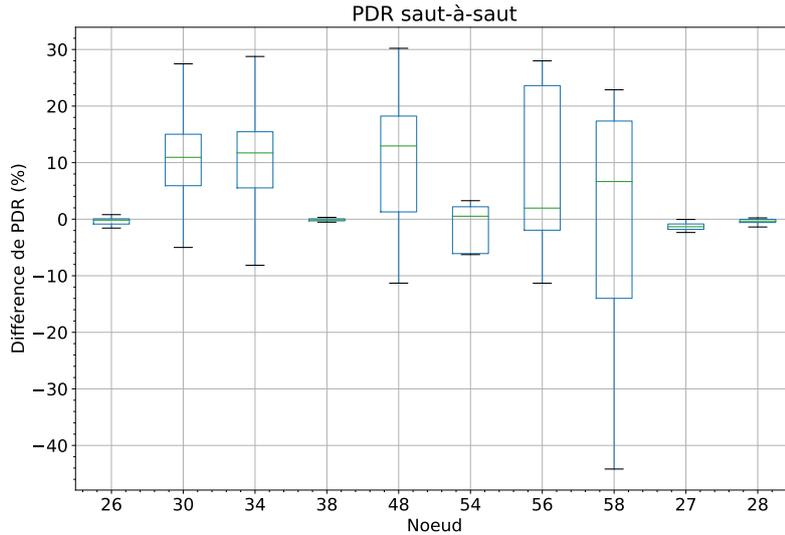


FIGURE 20 : Différence de PDR entre les 2 déclencheurs

#### 6.4 Améliorer le mécanisme de sélection des noeuds à rediriger

Les résultats des expériences précédentes étant concluants, nous avons également tenté d'améliorer le choix des noeuds à rediriger. Pour cela, nous avons encore ajouté une information supplémentaire aux DAO émis. Un noeud se signale comme prioritaire si le meilleur parent potentiel de l'instance voisine dispose d'un rang inférieur au parent actuel. Si les rangs sont équivalents, on compare l'ETX. Ainsi, lorsqu'un BR essaye de rediriger des noeuds, son mécanisme de sélection choisira d'abord les noeuds prioritaires lors de la redirection.

Nos premiers essais avec les paramètres de 6.3 et les pertes dans la file d'attente comme déclencheurs ont montré une légère diminution du nombre de transmissions et une légère augmentation du PDR. Toutefois, nous n'avons pas pu tester sur un nombre assez important d'expériences et la taille limitée du réseau ne permet pas d'obtenir des résultats concrets.

#### 6.5 Bilan des travaux expérimentaux

Les essais de RPL-NPT-LB se sont avérés concluants, nous constatons bien comme dans [10] une amélioration des performances globales du réseau en utilisant la même topologie. Nous avons également vu que ces résultats se transposent à des réseaux plus grands, où le gain est encore supérieur. De même, nous observons que l'ajout de BRs supplémentaires contribue largement à améliorer la qualité du réseau. Enfin, nous avons mis en lumière une des limites de l'utilisation de la taille du sous-DODAG d'un BR comme déclencheur du mode congestionné, celui-ci ne prenant pas en compte la congestion au niveau des noeuds enfants. L'utilisation d'un autre type de déclencheur comme le nombre de pertes dans la file d'attente dans un noeud enfant permet de prendre en compte cette situation, menant ainsi à une amélioration de la qualité du réseau, légère dans nos tests, mais néanmoins présente.

## 6.6 Perspectives

Les tests sur un déclencheur alternatif lié aux noeuds sont encourageants. Il serait toutefois intéressant de trouver une métrique différente qu'un nombre fixe de pertes dans la file d'attente, ou de le pondérer par d'autres valeurs. De même, détecter la congestion au niveau des noeuds enfants permet de s'adapter aux différences d'émissions des noeuds. En effet, dans un large réseau, tous les noeuds n'émettent pas forcément de manière uniforme. Il serait donc envisageable de tester des topologies plus larges avec des fréquences d'émission hétérogènes au sein du réseau. Par ailleurs, nous pourrions imaginer une façon d'utiliser plusieurs déclencheurs du mode congestionné en simultané, afin d'aboutir à des redirections les plus optimales possible.

En ce qui concerne la priorité lors de la redirection des noeuds, il serait intéressant vérifier le comportement avec des réseaux plus larges où il y a une grosse différence entre les rangs des noeuds. Dans un second temps, nous pourrions affiner la sélection parmi les noeuds prioritaires grâce à l'envoi de métriques. Les contraintes énergétiques n'ayant pas été traitées dans notre phase expérimentale, et l'une des prochaines étapes dans l'amélioration du protocole serait de s'intéresser à des expériences avec un duty cycle, en incorporant des métriques liées à l'énergie des noeuds dans les choix de redirection.

Enfin, une version plus récente de l'OS est maintenant activement développée, Contiki-NG [1], et introduit une nouvelle implémentation de RPL, RPL-Lite. Cette implémentation est plus légère, ne gérant qu'une partie réduite des fonctionnalités spécifiées dans la norme, et les limitations comme le support d'une seule instance RPL ou l'absence du mode stockant ne sont absolument pas problématiques pour RPL-NPT-LB. Il serait donc également intéressant de tester une implémentation du protocole sur cet OS.

## 7 Conclusion

Pour conclure, nous avons pu étudier les différentes contraintes liées à l'Internet des Objets. Pour rendre un Low-power and Lossy Network (LLN) tolérant à la panne et afin de le rendre plus durable et robuste, la multi-domiciliation du réseau via l'ajout de plusieurs routeurs de bordure (BR) est une solution fonctionnelle et efficace.

Nous avons cependant vu que les protocoles standards pour ce type de réseaux ne traitent pas de ce sujet. Il existe de nombreux travaux dans la littérature détaillant différents protocoles afin d'ajouter des BR supplémentaires dans un LLN, mais tous ne prennent pas en compte la totalité des problèmes liés à ce type de réseau, comme le traitement de la panne d'un BR ou l'utilisation de réseaux d'opérateurs différents. Nous avons présenté en détail deux solutions, Syn-RPL et RPL-NPT-LB, chacune ayant une approche spécifique (centralisée contre distribuée), mais toutes les deux tenant compte de tous les problèmes à résoudre.

Finalement, une phase expérimentale nous a permis de tester RPL-NPT-LB sur l'IoT-Lab et de vérifier son efficacité, d'étudier le comportement du protocole ainsi que de proposer des pistes d'amélioration.

Je tiens également à remercier Julien Montavont pour avoir proposé ce sujet qui m'a permis d'explorer des domaines inconnus pour ma part et qui deviennent des enjeux majeurs à notre époque, ainsi que pour son encadrement au cours du semestre.

## Références

- [1] Contiki-NG GitHub repository. <https://github.com/contiki-ng/>.
- [2] ContikiOS GitHub repository. <https://github.com/contiki-os/>.
- [3] FIT IoT-LAB : M3 Open Node. <https://www.iot-lab.info/hardware/m3/>.
- [4] *IEEE Std 802.15.4-2015 (Revision of IEEE Std 802.15.4-2011) : IEEE Standard for Low-Rate Wireless Networks*. IEEE, 2016.
- [5] ADJIH, C., BACCELLI, E., FLEURY, E., HARTEK, G., MITTON, N., NOEL, T., PISSARD-GIBOLLET, R., SAINT-MARCEL, F., SCHREINER, G., VANDAELE, J., AND WATTEYNE, T. FIT IoT-LAB : A large scale open experimental IoT testbed. In *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)* (Milan, Italy, Dec. 2015), IEEE, pp. 459–464.
- [6] CARELS, D., DERDAELE, N., POORTER, E. D., VANDENBERGHE, W., MOERMAN, I., AND DEMEESTER, P. Support of multiple sinks via a virtual root for the RPL routing protocol. *EURASIP Journal on Wireless Communications and Networking* 2014, 1 (Dec. 2014), 91.
- [7] DERU, L., DAWANS, S., OCAÑA, M., QUOITIN, B., AND BONAVENTURE, O. Redundant Border Routers for Mission-Critical 6LoWPAN Networks. In *Real-World Wireless Sensor Networks*, K. Langendoen, W. Hu, F. Ferrari, M. Zimmerling, and L. Mottola, Eds., vol. 281. Springer International Publishing, Cham, 2014, pp. 195–203.
- [8] FAROOQ, M. O., AND KUNZ, T. IoT-RF : A routing framework for the Internet of Things. In *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)* (Montreal, QC, Oct. 2017), IEEE, pp. 1–7.
- [9] FAROOQ, M. O., SREENAN, C. J., BROWN, K. N., AND KUNZ, T. RPL-based routing protocols for multi-sink wireless sensor networks. In *2015 IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)* (Abu Dhabi, United Arab Emirates, Oct. 2015), IEEE, pp. 452–459.
- [10] FOUBERT, B., AND MONTAVONT, J. Sharing is caring : a cooperation scheme for RPL network resilience and efficiency. In *2019 IEEE Symposium on Computers and Communications (ISCC)* (Barcelona, Spain, June 2019), IEEE, pp. 1–6.
- [11] GADDOUR, O., AND KOUBÂA, A. RPL in a nutshell : A survey. *Computer Networks* 56, 14 (Sept. 2012), 3163–3178.
- [12] GNAWALI, O., AND LEVIS, P. The Minimum Rank with Hysteresis Objective Function. Tech. Rep. RFC6719, RFC Editor, Sept. 2012.
- [13] HINDEN, R., AND HABERMAN, B. Unique Local IPv6 Unicast Addresses. Tech. Rep. RFC4193, RFC Editor, Oct. 2005.
- [14] HUI, J., AND THUBERT, P. Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks. Tech. Rep. RFC6282, RFC Editor, Sept. 2011.
- [15] JP. VASSEUR, KIM, M., PISTER, K., DEJEAN, N., AND BARTHEL, D. Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks. Tech. Rep. RFC6551, RFC Editor, Mar. 2012.

- [16] LEVIS, P., CLAUSEN, T., HUI, J., GNAWALI, O., AND KO, J. The Trickle Algorithm. Tech. Rep. RFC6206, RFC Editor, Mar. 2011.
- [17] MONTENEGRO, G., HUI, J., CULLER, D., AND KUSHALNAGAR, N. *Transmission of IPv6 Packets over IEEE 802.15.4 Networks*. No. 4944 in Request for Comments. RFC Editor, Sept. 2007.
- [18] NADAS, S. Virtual Router Redundancy Protocol (VRRP) Version 3 for IPv4 and IPv6. Tech. Rep. RFC5798, RFC Editor, Mar. 2010.
- [19] NGUYEN, Q.-D., MONTAVONT, J., MONTAVONT, N., AND NOËL, T. RPL Border Router Redundancy in the Internet of Things. In *Ad-hoc, Mobile, and Wireless Networks*, N. Mitton, V. Loscri, and A. Mouradian, Eds., vol. 9724. Springer International Publishing, Cham, 2016, pp. 202–214.
- [20] THUBERT, P. Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL). Tech. Rep. RFC6552, RFC Editor, Mar. 2012.
- [21] WASSERMAN, M., AND BAKER, F. IPv6-to-IPv6 Network Prefix Translation. Tech. Rep. RFC6296, RFC Editor, June 2011.
- [22] WINTER, T., THUBERT, P., BRANDT, A., HUI, J., KELSEY, R., LEVIS, P., PISTER, K., STRUIK, R., JP. VASSEUR, AND ALEXANDER, R. RPL : IPv6 Routing Protocol for Low-Power and Lossy Networks. Tech. Rep. RFC6550, RFC Editor, Mar. 2012.