



HAL
open science

Agrégation de trafic pour réduire la consommation énergétique globale dans les réseaux sans fil multi-sauts

Alexandre Laubé

► **To cite this version:**

Alexandre Laubé. Agrégation de trafic pour réduire la consommation énergétique globale dans les réseaux sans fil multi-sauts. Réseaux et télécommunications [cs.NI]. Université Paris Saclay (COmUE), 2017. Français. NNT : 2017SACLS298 . tel-01609917

HAL Id: tel-01609917

<https://theses.hal.science/tel-01609917>

Submitted on 4 Oct 2017

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

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

NNT : 2017SACLS298

THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PARIS-SACLAY
PRÉPARÉE À L'UNIVERSITÉ PARIS SUD

Ecole doctorale n°580
École doctorale des Sciences et Technologies de l'Information
et de la Communication (STIC)
Spécialité de doctorat : Réseaux, Information et
Communications

par

ALEXANDRE LAUBÉ

Agrégation de trafic pour réduire la consommation
énergétique globale dans les réseaux sans fil multi-sauts

Thèse présentée et soutenue à Orsay, le 25 Septembre 2017

Composition du Jury :

HOUDA LABIOD	Professeur Télécom ParisTech	(Présidente)
NADJIB ACHIR	Maître de conférences Université Paris 13	(Rapporteur)
NADJIB AIT SAADI	Professeur ESIEE Paris	(Rapporteur)
ANIS LAOUITI	Maitre de conférences Télécom SudParis	(Examineur)
GUY PUJOLLE	Professeur Université Pierre et Marie Curie	(Examineur)
SIDI MOHAMMED SENOUCI	Professeur Université de Bourgogne	(Examineur)
DOMINIQUE QUADRI	Maître de conférences Université Paris Sud	(Co-directrice de thèse)
STEVEN MARTIN	Professeur Université Paris Sud	(Directeur de thèse)

Remerciements

Je tiens à remercier particulièrement Steven Martin et Dominique Quadri, mes encadrants de thèse pour leur optimisme qui a grandement contribué à forger ma confiance envers mon travail et mes résultats, leur direction toujours à l'écoute et prête me remettre en question chaque fois que c'était nécessaire, et leur soutien tout au long de ces trois dernières années.

Merci à Khaldoun Al Agha pour m'avoir fait découvrir le monde de la recherche et sans qui j'aurais, sans doute, fait un autre choix de carrière.

Je tiens également à remercier les membres du jury, et en particulier Messieurs Achir et Ait Saadi d'avoir accepté de rapporter mon travail de thèse.

Merci à Hakima Chaouchi et Fabienne Lassausaie pour leur implication dans le programme de formation doctorale de l'EIT Digital sans quoi ce que je considère comme étant le point noir de ma thèse aurait pu être bien pire. Merci à Björn Landfeldt et son équipe pour l'accueil qu'ils m'ont offert lors de ma mobilité.

Et enfin, je remercie mes parents pour m'avoir toujours poussé dans mes études pré universitaires et m'avoir soutenu dans mon choix de poursuites d'étude.

Table des Matières

Remerciements	iii
Table des Matières	viii
1 Introduction	1
1.1 Contexte	1
1.2 Contributions	3
1.3 Organisation du document	5
2 État de l’art	7
2.1 Routage dans les réseaux sans fil multi-sauts	8
2.1.1 Protocoles de routage réactifs	9
2.1.2 Protocoles de routage proactifs	12
2.2 Contrôle de topologie	14
2.3 Extinction des nœuds	15
2.3.1 Duty Cycle	16
2.3.2 Systèmes Wake Up Radio (WuR)	16
2.4 Agrégation	17
2.5 Conclusion	18
3 Agrégation de flux	21
3.1 Notations et modèle de réseau étudié	23
3.1.1 Topologie de réseau	23
3.1.2 Modèle d’interférences	24
3.2 Programmation linéaire en nombres entiers	25
3.2.1 Présentation	25
3.2.2 Un exemple	26
3.2.3 Modélisation d’un problème de routage - Cas du plus court chemin (ILP)	28
3.3 Minimisation de la consommation énergétique	30
3.3.1 Modèle énergétique proposé	30
3.3.2 Programme d’optimisation - Minimisation de l’énergie (MinEnergy)	33
3.3.3 Simulation et résultats numériques	34
Outils de simulation	34
Paramètres et scénarii de simulation	36
Mesures	37

	Résultats de simulations	38
3.4	Minimisation du nombre de nœuds actifs	40
	3.4.1 Programme d'optimisation - Routage avec Agrégation de flux (RA)	41
	3.4.2 Résultats de simulation	42
3.5	Codage Réseau	45
	3.5.1 Introduction au codage réseau	45
	3.5.2 Application à l'agrégation de flux (RANC et RA+NC)	49
	3.5.3 Résultats des simulations	50
3.6	Conclusion	53
4	Métrique d'agrégation de flux : FAME	55
4.1	Métrique d'agrégation de Flux : FAME	56
	4.1.1 Intuition	56
	4.1.2 Nœuds d'intérêt	58
	4.1.3 Calcul de la métrique	59
	4.1.4 Résultats de simulation	61
4.2	Prise en compte des interférences	65
	4.2.1 Le problème des cliques	66
	4.2.2 Approximation des cliques	66
	4.2.3 Métrique d'agrégation de flux adaptative : A-FAME	67
	Hyperparamètre : seuil de réduction et base d'agrégation	68
	4.2.4 Résultats de simulation	69
4.3	Conclusion	71
5	Application	73
5.1	Rappel sur le protocole de routage OLSR	74
5.2	OLSR avec agrégation de flux (A-OLSR)	76
5.3	Simulations	78
	5.3.1 Network Simulator 3	78
	5.3.2 Simulations et résultats numériques	79
	Nouvelles mesures	80
	Résultats	80
5.4	Conclusion	85
6	Conclusion et Perspectives	87
	Publications	91

Chapitre 1

Introduction

1.1 Contexte

Dans les années 2000, le développement des réseaux suivait une logique de centralisation. L'arrivée sur le marché de services web et cloud tels que la recherche en ligne, les réseaux sociaux ou encore la vente en ligne ont concentré l'évolution de l'architecture des réseaux autour de centres de données capables d'amasser, de traiter et de distribuer de grandes quantités d'informations. La démocratisation des terminaux mobiles a contribué, quant à elle, au développement des réseaux sans fil basés sur des architectures de type point d'accès et donc à une centralisation aux niveaux physique et logique des réseaux. Si cette dynamique a permis de créer des applications réseau performantes et facilement gouvernables, les limites de ce type d'architecture commencent à se manifester. Les centres de données, fortement vulnérables à la perte de connexion et de données, doivent être dupliqués géographiquement un certain nombre de fois afin de garantir leur résilience. La concentration des données est également source de congestion, provoquant des baisses de performances dans les réseaux. La gestion des interférences et la réutilisation spatiale des fréquences est un des principaux problèmes rencontrés dans les réseaux mobiles. Enfin, le coût d'installation et d'entretien de ce genre d'infrastructures peut parfois limiter leur déploiement.

En parallèle, la communauté scientifique a réalisé de nombreux travaux portant sur les systèmes et réseaux distribués de type ad hoc. Dans un premier temps destinés à des cas d'utilisation bien particuliers, tels que la couverture de zones blanches, le déploiement rapide et temporaire en cas de gestion de sinistres ou l'utilisation de systèmes de diffusion de contenus distribués, la dynamique actuelle permet de donner un élan à ce domaine des réseaux et applications distribués. L'augmentation massive de l'utilisation des terminaux mobiles a permis la distribution des points d'accès au réseau cellulaire

1.1 Contexte

grâce au concept de femtocell, permettant une meilleure réutilisation spatiale des fréquences. Le *edge computing* commence à devenir une solution viable aux problèmes de délais et de débits en déportant au plus près des utilisateurs une partie des services qui sont aujourd'hui fournis par des centres de données très éloignés. Le *big data*, grande tendance de ces dernières années, contribue fortement à l'intérêt porté aux réseaux de capteurs. L'Internet des objets, les véhicules autonomes, la domotique, les villes intelligentes font partie de ces révolutions technologiques que les infrastructures centralisées actuelles seront incapables de supporter efficacement et qui favorisent l'émergence de systèmes distribués, diversifiés et plus adaptés aux contraintes d'aujourd'hui en matière d'infrastructure, de considérations sociales et d'environnement.

Cette dernière contrainte est par ailleurs au cœur de nombreux travaux dans le domaine des réseaux. La volonté de tendre vers une société "tout électrique" apporte davantage de crédibilité à la crainte de voir un jour arriver un pic énergétique où la demande deviendra supérieure à l'offre. De nombreuses recherches sont ainsi entreprises dans le domaine de la production, du stockage et, dans le cadre des technologies de l'information, de l'utilisation de l'énergie afin de repousser toujours plus l'échéance. En informatique, de nombreux efforts ont été accomplis au niveau matériel afin de réduire la consommation énergétique des composants électroniques. Des considérations énergétiques peuvent également être prises au niveau applicatif en réduisant la complexité des algorithmes ou encore au niveau réseau en adaptant les protocoles de communication afin d'améliorer leur efficacité énergétique.

C'est dans cette dynamique que s'inscrit cette thèse. Partant du postulat que la consommation énergétique d'un nœud d'un réseau est, finalement, beaucoup moins influencée par son activité en termes de transmission et de réception que par le simple fait d'être mis sous tension ou non, cette thèse vise à maximiser le nombre de nœuds pouvant potentiellement être éteints, ou mis en veille. Ceci a pour but de réduire la consommation énergétique globale du réseau plutôt qu'au niveau local et individuel des nœuds, grâce à une méthode de routage permettant d'agrèger les flux d'un réseau sur un minimum de nœuds. Sans remettre en cause la qualité de service offerte aux applications en termes de débit, une telle approche permet à un algorithme d'extinction de nœuds d'éteindre ceux qui ne sont pas utilisés. Cette solution va à contre courant de ce qui est généralement fait dans les réseaux sans fil multi-sauts, où la plupart des travaux considèrent que les nœuds sont énergétiquement contraints et doivent donc se partager la charge du réseau afin de consommer leur réserve d'énergie équitablement.

1.2 Contributions

Cette thèse s'intéresse au concept d'agrégation de flux dans les réseaux sans fil multi-sauts, c'est-à-dire au fait d'acheminer du trafic réseau en utilisant le moins de nœuds possible. Cette démarche a pour but d'offrir un moyen d'améliorer la réduction de la consommation énergétique globale des réseaux sans fil multi-sauts grâce à l'extinction, ou la mise en veille, des nœuds. Ainsi, trois principales contributions, autour desquelles s'articule par ailleurs ce manuscrit, ressortent de cette thèse, auxquelles viennent s'ajouter deux autres plus contextuelles mais tout aussi importantes.

Les **principales contributions** portent sur l'agrégation de flux, avec prise en compte de la qualité de service en termes de débit. Router des flux dans un réseau peut être fait soit de façon statique, en déterminant une fois pour toute le chemin que doit emprunter un flux pour aller du nœud source au nœud destination, soit de façon dynamique, en utilisant un algorithme de routage de plus court chemin où les nœuds se voient affectés des poids. Le faire de manière impartiale et équitable s'avère souvent être un problème d'optimisation difficile au vu des contraintes à respecter, dès lors qu'il s'agit de réseaux sans fil multi-sauts. Le routage avec agrégation de flux ajoute à cette complexité, à cause de l'interdépendance entre les flux. En effet, le chemin à sélectionner entre deux nœuds peut ne pas être le même en fonction des flux présents dans le réseau.

- La première contribution de cette thèse consiste à résoudre ce problème d'optimisation. Pour cela, nous proposons une solution basée sur la programmation linéaire en nombres entiers permettant d'agréger au maximum les flux, c'est-à-dire minimisant le nombre de nœuds nécessaires pour router un ensemble donné de flux. Cette agrégation optimale respecte la qualité de service des flux en termes de débit. Pour évaluer les performances de notre solution, nous avons développé un outil de simulation s'appuyant sur des modèles théoriques de graphes, d'interférences et de consommation énergétique. Nous montrons la corrélation entre la consommation énergétique globale du réseau et le nombre de nœuds pouvant potentiellement être éteints, ainsi que l'impact de l'agrégation de flux sur la capacité du réseau. Notre solution étant optimale par définition, elle permet de disposer d'une borne supérieure quant au gain maximal atteignable en matière d'économie d'énergie, si les nœuds non utilisés devaient être éteints ou mis en veille. Cette borne servira de référence pour évaluer les autres contributions.

1.2 Contributions

- La seconde contribution propose une solution permettant l'application de l'agrégation de flux de façon distribuée. En effet, la solution optimale n'est pas applicable en pratique car centralisée et complexe. Nous proposons donc une métrique de routage pour des algorithmes génériques de plus court chemin. Cette solution s'appuie sur la définition de nœuds d'intérêt et de la position d'un nœud par rapport à ces derniers. Il est montré par simulation que la métrique proposée offre des performances très proches de la solution optimale proposée précédemment.
- La troisième contribution consiste en l'intégration de notre métrique d'agrégation dans un protocole de routage proactif qu'est OLSR. Après avoir décrit une implémentation possible, des simulations réalisées sous le simulateur de réseau NS-3 montrent l'efficacité de la métrique en termes d'agrégation et de gain énergétique. Les résultats permettent également de voir que le protocole OLSR, de par sa nature à réduire la connectivité du réseau au moyen de son mécanisme de MPRs, ne permet pas de profiter pleinement de la puissance de la métrique d'agrégation. Toutefois, ils sont très encourageants et mettent en lumière la nécessité d'étendre le mécanisme de sélection des MPRs pour tendre vers des performances quasi-optimales.

L'agrégation de flux permet d'obtenir une meilleure réduction de la consommation énergétique lorsque les nœuds non utilisés sont éteints. Cependant, cette méthode n'est pas exempte de limites qui peuvent dans certains cas devenir problématiques. L'une d'elles concerne la concentration du trafic. En agrégeant les flux entre eux, les nœuds qui servent à router ces derniers voient leur charge augmenter. De plus, agréger les flux a un coût en termes d'augmentation de la longueur moyenne des chemins par rapport à une solution de routage basée sur du plus court chemin. La combinaison de ces deux facteurs a pour effet d'augmenter la charge globale du réseau, ce qui peut occasionner une baisse des performances.

- Nous résolvons le problème d'augmentation de charge du réseau en étudiant l'interaction entre l'agrégation de flux et le codage réseau inter-flux. Le codage réseau définit un concept de retransmission dans lequel les nœuds intermédiaires d'un réseau vont combiner les paquets de plusieurs flux avant de retransmettre ces combinaisons. Ainsi, le nombre de transmissions globales est réduit. Ceci a pour effet de diminuer la charge du réseau, parfois même en-deçà de la charge obtenue avec un routage de plus court chemin. La synergie entre l'agrégation

de flux et le codage réseau est pertinente, la première technique permettant de créer des opportunités pour le codage réseau et la seconde pour permettre une plus forte agrégation. Il est également possible d'appliquer du codage réseau indépendamment de l'agrégation. Cela permet de pouvoir implémenter facilement les deux mécanismes. Nous montrons que cette dernière solution est la plus avantageuse.

- Enfin, la dernière contribution propose une simplification du modèle d'interférences à base de cliques afin de permettre à l'agrégation de flux de s'adapter à la charge du réseau dans le cas de son application dans un environnement distribué. Les résultats montrent que cette méthode permet à l'algorithme de routage d'agréger au mieux les flux lorsque le réseau est peu chargé, puis de passer progressivement vers un routage sur le plus court chemin en nombre de sauts lorsque le trafic augmente. L'objectif ici est de permettre à l'algorithme de routage d'agréger les flux en prenant en compte les interférences de façon dynamique et distribuée avec une complexité réduite afin de répondre aux contraintes de délai nécessaires au bon fonctionnement du protocole de routage.

1.3 Organisation du document

Les chapitres de ce document sont organisés de façon à construire itérativement, au fur et à mesure de la lecture, la base de connaissances qui a permis de réaliser les différentes étapes du travail accompli pendant cette thèse. Ainsi, chaque chapitre (à l'exception du premier) et chaque section introduit une nouvelle notion qui viendra s'ajouter à celles précédemment présentées et qui sera un support pour les travaux présentés plus tard dans ce document, ou qui fera l'objet d'une étude visant à résoudre un problème complémentaire au sujet considéré.

Le chapitre 2 présente une revue des différentes approches proposées par la communauté scientifique en matière de réduction de la consommation énergétique et de routage dans les réseaux sans fil multi-sauts et permet de situer le travail de cette thèse au sein de cette dynamique.

Le chapitre 3 constitue le cœur du travail réalisé durant cette thèse et décrit le cheminement, étape par étape, qui a conduit à l'élaboration et à l'étude du concept de routage avec agrégation de flux au sens entendu dans cette thèse.

1.3 Organisation du document

Le chapitre 4 laisse entrevoir la faisabilité du routage avec agrégation de flux en décrivant une solution heuristique sous-optimale, efficace, moins complexe et pouvant être utilisée dans des algorithmes de routage distribués. Deux versions de cette heuristique sont proposées, dont une capable de prendre en compte les interférences et de s'adapter à la charge du réseau.

Le chapitre 5 aborde l'étude de l'agrégation de flux dans un cadre semi-concret en intégrant l'une des heuristiques présentées dans le chapitre précédent à un protocole de routage existant. Cette expérimentation permet d'illustrer l'applicabilité de l'agrégation de flux et met en évidence certaines des difficultés qui peuvent être rencontrées lors du passage de la théorie à la pratique.

Enfin, une conclusion viendra résumer le travail présenté dans ce document et abordera plusieurs perspectives de recherche liées aux solutions proposées.

Chapitre 2

État de l'art

Dans une société où la tendance est au tout électrique avec le développement de l'Internet des objets, des véhicules électriques, des panneaux publicitaires numériques ou encore des services de big data, des efforts doivent être faits afin d'éviter une explosion de la demande en énergie. À ce jour, beaucoup de travaux ont été entrepris au niveau des composants électroniques pour réduire efficacement la consommation énergétique des nœuds d'un réseau. Toutefois, des gains significatifs peuvent encore être obtenus en se focalisant sur l'aspect protocolaire des réseaux. Dans les réseaux sans fil multi-sauts en particulier, plusieurs solutions ont été proposées ces dernières années pour réduire davantage la consommation énergétique.

Les principales contributions portent sur le routage, avec des extensions aux protocoles existants pour prendre en compte cette problématique. En effet, dans la plupart des cas, des solutions efficaces de routage existent pour ce type de réseaux. De nombreux travaux ont donc été menés pour les rendre économes en énergie, mais généralement en cherchant une optimisation individuelle de la consommation énergétique des nœuds afin que ceux-ci, alimentés sur batterie, puissent fonctionner le plus longtemps possible. L'état de l'art n'apporte donc pas de solution pour une minimisation de la consommation d'énergie du réseau dans son ensemble. Or cette problématique est importante à prendre en compte, en particulier lorsque les nœuds du réseau ne sont pas contraints en termes d'alimentation (batteries ayant des durées de charge importantes, source d'énergie continue).

D'autres approches se sont focalisées sur le contrôle de puissance des nœuds, permettant de modifier leur puissance de transmission afin de réduire leur consommation énergétique lors de l'envoi des données. Faire varier la puissance de transmission a un impact sur la distance de réception et donc sur la topologie dans les réseaux sans fil

2.1 Routage dans les réseaux sans fil multi-sauts

multi-sauts. Le terme employé dans ce cas est alors contrôle de topologie. Dans ce domaine, les principaux travaux portent sur l'amélioration des performances en termes de débit, en maximisant notamment la réutilisation spatiale. Peu d'études se sont intéressées aux aspects énergétiques.

Un autre axe de recherche s'est intéressé à la consommation, consistant à affecter aux nœuds une politique d'économie d'énergie basée sur leur comportement. Par exemple, réduire la fréquence de l'horloge du processeur peut permettre de réduire sa consommation énergétique, limitant naturellement les performances (moins d'instructions exécutées à la seconde). Dans les réseaux sans fil multi-sauts, le *Duty Cycle* illustre ce paradigme en permettant aux nœuds de passer périodiquement d'un état actif de travail à un état de veille. Cette approche est défendue par un modèle se basant sur des états énergétiques [1, 2], un nœud (ou son interface réseau sans fil) pouvant être dans plusieurs états d'activité différents au cours du temps, chacun ayant sa propre consommation.

Enfin, l'agrégation est une technique parfois utilisée pour améliorer l'efficacité énergétique des réseaux, mais est souvent redéfinie et/ou associée à d'autres notions afin de remplir les objectifs des algorithmes dans lesquels elle est utilisée. Nous apportons donc notre contribution à ce type d'approche en proposant l'agrégation de flux qui n'a, à notre connaissance, jamais été utilisée telle que nous le faisons dans cette thèse.

La première partie de ce chapitre rappelle les protocoles de routage dans les réseaux sans fil multi-sauts et leurs extensions apportant des fonctionnalités d'économie d'énergie. La deuxième partie s'intéresse au cas du contrôle de topologie. La troisième partie présente les techniques de réduction de la consommation énergétique se basant sur l'extinction des nœuds. La quatrième partie aborde l'utilisation de mécanismes d'agrégation dans les réseaux sans fil multi-sauts et introduit notre définition de l'agrégation de flux. Enfin, une conclusion résume les caractéristiques des différentes solutions et présente la position prise vis-à-vis de la problématique traitée dans cette thèse.

2.1 Routage dans les réseaux sans fil multi-sauts

Par nature, le routage dans les réseaux sans fil multi-sauts se base sur des algorithmes distribués. Étant donné que le déploiement de ce type de réseaux ne nécessite pas l'installation d'une infrastructure centralisée, chaque nœud le composant partage les mêmes rôles de relai et d'utilisateur. S'il existe aujourd'hui des méthodes pour per-

mettre de déployer facilement des applications centralisées telles que les réseaux virtuels type "overlay" ou SDN (Software Defined Networks) [3, 4], ces approches nécessitent une connaissance globale du réseau qu'il est difficile de fournir sans dégrader la capacité de ce dernier. En effet, chaque changement de l'état du réseau (topologie, trafic, interférence, ...) nécessite l'envoi d'une information de mise à jour au contrôleur central qui recalculera ensuite les informations de routage à diffuser en retour à l'ensemble du réseau.

Pour éviter ce problème, les algorithmes de routage distribués implémentent des protocoles "légers" en termes de messages envoyés et placent le calcul des tables de routage au sein de chaque nœud. Nous pouvons distinguer deux types d'algorithmes de routage distribués :

Routage réactif Un nœud initie l'algorithme de calcul des routes au moment où il a un paquet à envoyer vers une destination pour laquelle il ne connaît pas encore de chemin. Ce type de protocole dispose d'un coût en message extrêmement réduit et s'avère très efficace quand il est utilisé dans des réseaux de grande taille. En revanche, le délai induit par le calcul des routes rend ce type d'algorithme inefficace, en particulier lorsque sont considérées des applications pour lesquelles le délai est un critère de Qualité de Service (QoS).

Routage proactif Un nœud va régulièrement actualiser ses informations de routage en fonction des messages de mise à jour qui lui sont adressés par ses voisins. Ainsi, lorsqu'un nœud souhaite envoyer un paquet à une destination, la route est déjà connue et permet de résoudre le problème de délai des protocoles de routage réactifs. Toutefois, cette amélioration a un coût non négligeable en termes de transmissions de données. En effet, la plupart de ces protocoles utilisent des algorithmes de découverte de topologie afin de permettre aux nœuds de construire localement une topologie du réseau, complète ou partielle, à partir de laquelle ils peuvent calculer les routes vers tous les autres nœuds. Ces protocoles sont généralement efficaces pour des réseaux denses et permettent aux nœuds d'avoir une vision de l'état actuel du réseau.

2.1.1 Protocoles de routage réactifs

Dynamic Source Routing [5] (DSR) et Ad hoc On-Demand Distance Vector [6] (AODV) Routing sont les deux principaux protocoles de routage réactifs et représentent deux approches de routage différentes. DSR applique du routage à la source. Plus précisément, le nœud source (le nœud souhaitant envoyer des données) fixe le

2.1 Routage dans les réseaux sans fil multi-sauts

chemin à suivre pour atteindre la destination (le nœud devant recevoir les données). AODV, en revanche, utilise une table de routage. Cette table permet à un nœud souhaitant envoyer ou devant retransmettre des données vers une destination, de savoir le voisin vers lequel les transférer afin d'atteindre le nœud cible. Ce voisin est couramment appelé le prochain saut à suivre pour atteindre une destination donnée. Dans le cas où les informations de routage (chemin complet ou prochain saut) sont inconnues, la source initie un algorithme de découverte de chemin vers sa destination. Cet algorithme est similaire pour les deux protocoles. L'initiation de l'algorithme se fait par la diffusion d'un message de type Route Request (RREQ). Lorsque qu'un nœud reçoit un RREQ pour une destination en particulier, si ce nœud n'est pas la destination, alors il insère dans le message son identifiant afin de mettre à jour le chemin déjà parcouru par le paquet et le retransmettre à tous ses voisins. Lorsque la destination reçoit un RREQ, elle renvoie un message de réponse (RREP) vers la source, qui suit le chemin inverse du RREQ. Durant cette étape, les nœuds font suivre le RREP jusqu'à la source afin qu'elle puisse sauvegarder le chemin découvert. Avec AODV, les nœuds vont, en plus de retransmettre le RREP jusqu'à la source, inspecter le chemin contenu dans le message afin de remplir leur table de routage avec le prochain saut vers la destination. Enfin, pour chacun des nœuds, des mécanismes permettent de sélectionner le plus court chemin. Si ces algorithmes sont efficaces pour calculer les meilleurs chemins reliant deux nœuds, il n'ont toutefois pas été définis en considérant la consommation énergétique et ne sont donc pas adaptés à des réseaux où ce critère est important.

C'est pourquoi des extensions ont été proposées pour ces protocoles afin de leur apporter des fonctionnalités en termes de consommation d'énergie. Dans [7], l'état énergétique d'un nœud permet de changer son comportement vis-à-vis du routage. Les auteurs définissent dans AODV trois états dans lesquels un nœud peut se trouver en fonction de son énergie résiduelle (charge restante). Un nœud est considéré comme étant dans un état normal si son énergie résiduelle est au-dessus d'un seuil prédéfini. Sinon, le nœud est dans un état d'alerte, voire dans un état de danger si son énergie résiduelle passe sous un seuil critique.

- Dans l'état normal, le nœud va suivre classiquement les spécifications de AODV.
- Dans l'état d'alerte, le nœud va ajouter un délai avant de faire suivre les RREQs, de sorte que les chemins passants par ce nœud deviennent moins intéressants dans l'estimation des meilleurs chemins.
- Dans l'état de danger, les RREQs sont bloqués afin de retirer ce nœud des potentiels relais du réseau.

Un mécanisme de Duty Cycle, qui sera présenté dans la partie 2.3.1, permet également d'éteindre temporairement les interfaces réseau des nœuds non utilisés pour une meilleure économie d'énergie. Dans [8], la durée de vie du réseau est améliorée par une sélection des chemins en fonction du délai de bout-en-bout et de la consommation énergétique. Plus précisément, le protocole AODV est modifié de sorte que la destination attende de recevoir plusieurs RREQs avant d'envoyer un RREP le long du chemin sélectionné. La durée de vie d'un chemin est calculée en fonction de l'énergie résiduelle minimale parmi les nœuds du chemin. En utilisant la valeur d'énergie résiduelle des nœuds, la notion de drain d'énergie des nœuds et l'estimation de la durée des sessions de flux, il est possible de sélectionner les chemins les plus à même de satisfaire dans leur totalité les flux entrants dans le réseau [9]. Une procédure proactive permet également d'éviter la rupture de la communication entre la source et la destination en calculant un chemin alternatif lorsque l'énergie résiduelle d'un nœud devient faible.

Cette approche montre des améliorations en termes de durée de vie du réseau et de QoS. Dans [10], la solution proposée intègre du contrôle de puissance de transmission avec des considérations énergétiques dans le protocole AODV. La destination répond de manière systématique à chaque RREQ et étend les messages RREPs avec un champ contenant l'énergie résiduelle des nœuds intermédiaires afin de sélectionner le chemin ayant la meilleure autonomie. Le contrôle de puissance intervient au niveau des nœuds intermédiaires au moment de faire suivre un paquet au prochain saut. Lorsqu'un nœud reçoit un paquet, il calcule la puissance de réception de ce dernier et renvoie cette valeur à l'émetteur pour que celui-ci puisse ajuster sa puissance de transmission en conséquence pour tous les autres paquets à suivre. Enfin, dans [11], les auteurs définissent une méthode d'optimisation bi-objectif afin de calculer une métrique de routage en fonction de la stabilité des routes et de l'énergie résiduelle des nœuds. De nouveau, cette approche vise à améliorer la durée de vie du réseau tout en fournissant des améliorations en termes de stabilité et de taux de paquets délivrés, ou PDR (Packet Delivery Ratio).

Concernant les travaux d'extension du protocole DSR, nous pouvons citer la solution proposée dans [12], qui utilise une approche similaire à [7] où les nœuds ayant une énergie résiduelle faible vont commencer à ajouter du délai dans la retransmission des RREPs. Cette approche a toutefois la particularité de rendre ce délai inversement proportionnel à la durée de vie estimée du nœud. Ainsi, plus l'énergie résiduelle d'un nœud diminue, plus le délai augmente, rendant les chemins passant par ce nœud in-

2.1 Routage dans les réseaux sans fil multi-sauts

appropriés pour l'algorithme de routage. Toujours en se basant sur l'énergie résiduelle des nœuds, [13] propose une solution améliorant la durée de vie du réseau, les délais de bout-en-bout et la dissipation énergétique (le partage équitable de la consommation d'énergie entre les nœuds). Dans [14], les auteurs proposent un système de contrôle d'admission des flux se basant sur l'énergie et considèrent la possibilité pour les nœuds d'avoir accès à un système de recharge de batterie (panneau solaire, générateur hydraulique, ...). Bien que ce travail soit présenté avec une application dans DSR, la métrique qui en résulte peut aussi être intégrée à des algorithmes de routage proactifs. Enfin, [15] propose une extension s'appuyant sur une approche multi-chemins de DSR, en considérant l'énergie résiduelle des nœuds et leur taux de drain énergétique pour calculer les deux routes disjointes ayant la durée de vie la plus élevée.

2.1.2 Protocoles de routage proactifs

Le protocole Optimized Link State Routing [16] (OLSR) est l'un des protocoles de routage proactifs les plus populaires. Ce protocole définit plusieurs messages qui sont régulièrement diffusés dans le réseau afin de réaliser des tâches de découverte de topologie. L'algorithme de diffusion est amélioré grâce à l'utilisation de Relais Multi-Points (MPR), seuls nœuds autorisés à faire suivre des paquets durant les phases de diffusion. Les MPRs sont élus par les autres nœuds afin de leur permettre d'atteindre leur voisinage à deux sauts. Les messages HELLO et TC (Topology Control) sont les deux principaux messages nécessaires au protocole pour que les nœuds puissent calculer leur table de routage. Ainsi, chaque nœud diffuse périodiquement un message HELLO afin de notifier sa présence à ses voisins. Ces messages ne sont jamais retransmis au-delà du voisinage à un saut du nœud émetteur. Ils permettent de signaler les liens et de savoir s'ils sont ou non bidirectionnels. Un nœud recevant un message HELLO peut avoir une connaissance de son voisinage à deux sauts et de la nature des transmissions possibles avec ce dernier. Les messages TC sont des messages d'état de liaison, générés et diffusés par les MPRs afin d'informer l'ensemble du réseau de l'existence de leurs liens symétriques avec leurs voisins à un saut. Les nœuds recevant ces messages peuvent alors (re)construire un graphe partiel de la topologie du réseau et calculer leur table de routage vers les destinations connues. Mais comme pour AODV et DSR, la spécification initiale du protocole OLSR ne prend pas en compte les aspects énergétiques du réseau.

Cependant, le protocole OLSR est conçu pour permettre l'intégration d'extensions. Ainsi, il est proposé dans [17] de router des flux avec des critères de QoS, en définissant un nouveau type de MPRs (nommés QMPRs). La solution étend les messages HELLO et TC en ajoutant un label de QoS (bande passante disponible, délai, puissance de

transmission, taux de perte, consommation énergétique, etc.) aux liens annoncés. Bien que les routes puissent ensuite être calculées de sorte à améliorer la qualité de service, cette spécification n'encourage pas l'économie d'énergie. Dans [18], les auteurs proposent une métrique se basant sur l'énergie résiduelle des nœuds et leur consommation pour modifier l'algorithme de sélection des MPRs. Puisque les MPRs servent de relais lors de la transmission des paquets, cette méthode tend à sélectionner des relais proposant une plus grande durée de vie. Un processus permet également aux nœuds de détecter les transmissions unicast qui ne leur sont pas destinées et d'éteindre leur interface réseau afin d'éviter de consommer l'énergie induite par la réception et le traitement de ces transmissions indésirables. Enfin, l'algorithme de routage permet de sélectionner les chemins offrant un coût énergétique minimal. Les fonctions de coût utilisées en général dans les réseaux impliquent les métriques MTPR (Maximum Transmission Power Routing) [19], MMBCR (Max-Min Battery Cost Routing) [20] et MDR (Minimum Drain Rate) [21].

Les résultats présentés montrent de bonnes performances vis-à-vis de la durée de vie du réseau, mais ne donnent rien concernant la consommation énergétique globale du réseau. Dans [22], les auteurs proposent de sélectionner les MPRs en fonction de la valeur réelle de l'énergie résiduelle et calculent les chemins en ayant le maximum. Cette approche permet une modeste amélioration en termes de PDR mais n'apporte aucune conclusion sur la consommation énergétique. Dans [23], les auteurs associent l'énergie résiduelle et la consommation énergétique des transmissions afin d'estimer la réserve d'énergie des nœuds dans le futur proche. Mais les résultats obtenus, qui montrent une amélioration vis-à-vis de la consommation énergétique en fonction de la mobilité (vitesse de déplacement des nœuds), ignorent l'impact de cette dernière sur le calcul des routes. La mobilité pouvant détruire des routes, des messages peuvent ne plus avoir de chemin à suivre et le nombre de transmissions dans le réseau se voit réduit significativement. En 2014, [24] propose une nouvelle version du protocole OLSR. OLSRv2 dispose désormais de deux types de MPRs. Les uns permettent d'optimiser les phases de diffusion des paquets pour les besoins du protocole (plan de contrôle), les autres servent à définir la topologie à utiliser pour les communications des applications (plan de données). Cette extension intègre également nativement la possibilité de diffuser des métriques, standardisant ainsi leur utilisation dans le protocole. Enfin, il est proposé dans [25] de combiner OLSR à une méthode de clustering en utilisant des métriques basées sur l'énergie. Toutefois, les résultats ignorent aussi l'aspect énergétique du réseau pour se concentrer sur les performances de clustering de l'algorithme présenté, en fonction de la mobilité du réseau.

2.2 Contrôle de topologie

Le contrôle de topologie est un domaine largement étudié dans le but de réduire les interférences, améliorer les performances ou encore réduire la consommation énergétique des réseaux. Dans ce dernier cas, les travaux existants visent généralement à augmenter la durée de vie des réseaux de capteurs. Ainsi, [26] propose une solution permettant de réduire le coût énergétique des transmissions des paquets dans les réseaux de capteurs. Les nœuds créent de manière distribuée une topologie de sorte que leurs voisins directs soient atteignables avec une puissance de transmission minimale. La métrique MTPR [19] est utilisée pour le calcul des routes et un protocole de type Duty Cycle permet de réduire davantage la consommation énergétique des nœuds. Toutefois, cette solution se base sur la position géographique des nœuds pour établir la topologie du réseau, ce qui la rend moins intéressante pour des réseaux mobiles.

Dans [27], les nœuds sont configurés pour utiliser un ensemble fini de puissances de transmission et calculent la topologie correspondant à la puissance de transmission minimale commune à tous. Pour mettre en œuvre cette solution, les nœuds doivent donc calculer toutes les topologies relatives aux différentes puissances de transmission disponibles, ce qui nécessite beaucoup de ressources. Dans [28], il est proposé de construire des clusters de nœuds connectés par un *connected dominating set* (CDS) [29], afin de créer un réseau cœur (backbone). Les nœuds peuvent ensuite choisir de rejoindre le backbone du réseau ou de se mettre en mode veille jusqu'à ce qu'ils aient des données à transmettre. Une fonction de rotation permet de changer les nœuds qui composent le backbone et ainsi améliorer la dissipation d'énergie du réseau.

Une autre approche utilisant des clusters de nœuds est introduite dans [30]. Ici, les nœuds n'élisent pas de leader (clusterhead) ou de passerelle. Ils sélectionnent une puissance de transmission qui est ensuite utilisée pour les communications intra-cluster. Le routage est alors calculé en utilisant la métrique MTPR. Enfin, dans [31], les auteurs considèrent que les nœuds peuvent modifier dynamiquement leur puissance de transmission afin de changer la topologie du réseau à la demande. Ils utilisent des informations de trafic (demandes des flux) et un modèle d'interférence à N-sauts pour maximiser le taux d'acceptation des flux. Les résultats montrent de très bonnes performances en termes de charge et de débit, mais ne considèrent pas la consommation énergétique.

2.3 Extinction des nœuds

Éteindre un appareil électronique est un moyen simple et efficace de réduire sa consommation énergétique. Pour les nœuds des réseaux sans fil multi-sauts, cette approche est défendue par un modèle énergétique se basant sur des états énergétiques [1, 2]. Un nœud, ou son interface réseau sans fil, peut être dans plusieurs états d'activité différents au cours du temps, chacun ayant son propre niveau de consommation. Dans les réseaux, quatre états énergétiques sont généralement utilisés, à savoir :

- **Mode Transmission (Tx)** Un nœud est dans cet état lorsqu'il transmet des paquets. Il s'agit classiquement de l'état dans lequel le nœud consomme le plus.
- **Mode Réception (Rx)** Pour un nœud, cet état est synonyme de réception d'un paquet qui lui est destiné. La consommation est considérée plus faible que dans l'état précédent, le nœud étant un peu plus passif, malgré l'activation des circuits pour décoder le paquet et envoyer les données à la couche applicative.
- **Mode Attente (Idle)** Il s'agit généralement de l'état dans lequel un nœud passe le plus de temps, c'est-à-dire lorsque qu'il ne transmet ni ne réceptionne de paquet. Le nœud est en attente d'avoir des données à envoyer ou à recevoir.
- **Mode veille (Sleep)** C'est l'état dans lequel un nœud consomme le moins d'énergie. En mode veille, il ne peut ni transmettre ni recevoir de données, ni même détecter l'arrivée d'un nouveau paquet. Cet état ne peut être quitté qu'en réponse à un signal de réveil (interruption utilisateur ou algorithmique).

L'objectif des méthodes d'économie d'énergie Duty Cycle [32] est de maximiser le temps que passe un nœud dans l'état Sleep tout en réduisant la dégradation de la QoS du réseau. En effet, plus un nœud passe de temps dans cet état, plus sa consommation moyenne diminue, mais moins il est réactif aux sollicitations du réseau (changements topologiques, transmissions de données, etc.).

De récents et intéressants travaux proposent de résoudre ce problème en intégrant dans les nœuds deux interfaces différentes de communication sans fil. Les systèmes Wake Up Radio (WuR) [33] associent une antenne de communication sans fil "classique", assurant la transmission des données dans le réseau, à une seconde fonctionnant à très faible puissance et ayant une capacité de transfert très faible. Cette seconde interface sans fil est utilisée par les nœuds pour envoyer un signal de réveil à leurs voisins en veille. Cette approche offre ainsi un bon compromis entre le mode Idle qui consomme davantage et le mode Sleep qui manque de réactivité. Elle reste cependant contraignante à mettre en place.

2.3.1 Duty Cycle

Le Duty Cycle est une méthode d'économie d'énergie permettant aux nœuds d'un réseau de planifier des périodes d'activité et de mise en veille. Dans [34], les auteurs améliorent la durée de vie du réseau en éteignant les nœuds considérés comme redondants. En utilisant la position géographique des nœuds, une grille virtuelle est créée de sorte que chaque nœud d'une cellule puisse communiquer avec ceux des cellules adjacentes. Un algorithme de rotation permet ensuite d'élire le nœud devant rester allumer de manière à répartir la consommation énergétique entre tous les occupants de la cellule. Cette approche a toutefois l'inconvénient d'utiliser la position géographique des nœuds et n'est donc applicable qu'à un nombre restreint de cas.

Dans [35], le nombre de nœuds en veille dans le réseau est maximisé en intégrant un court signal de notification afin qu'ils puissent rapidement détecter si un message leur est destiné à leur sortie de veille. De cette manière, les nœuds peuvent rapidement se remettre en mode veille le cas échéant. Cependant, cette technique ne résout pas le problème de réactivité des nœuds devant se réveiller.

Un autre algorithme est proposé dans [36], construisant un backbone afin d'éteindre les nœuds n'en faisant pas partie. Un processus d'évaluation des performances permet ensuite de rallumer un nœud et de l'ajouter au backbone si besoin, selon la QoS demandée par la couche applicative. Mais ce calcul se faisant au réveil des nœuds, le problème de réactivité n'est toujours pas résolu.

2.3.2 Systèmes Wake Up Radio (WuR)

Le problème principal des méthodes d'économie d'énergie se basant sur le Duty Cycle est que leur réactivité aux événements du réseau dépend de leurs paramètres de configuration. Par exemple, définir une longue période de sommeil pour une économie maximale impacte significativement la réceptivité du nœud, que ce soit pendant la phase de sommeil alors que des messages lui sont destinés ou dans un autre mode alors que le réseau est dans un moment d'inactivité. De plus, ce comportement peut avoir des effets très négatifs sur les protocoles exécutés, en particulier en termes de routage. Pour palier ce problème, il est proposé dans [37] d'augmenter la durée de vie des réseaux de capteurs en intégrant aux capteurs un deuxième récepteur sans fil. Ce nouveau composant a la particularité de pouvoir extraire l'énergie contenue dans un signal afin d'envoyer une instruction de réveil au nœud. Toutefois, ce type de dispositif nécessite l'envoi d'un signal suffisamment puissant pour compenser l'atténuation causée

par la distance de transmission. Cette application est donc, en plus d'être limitée par une architecture particulière (système d'extraction d'énergie), restreinte à des réseaux très denses avec une grande proximité entre les nœuds.

Des travaux plus récents ont fait le choix d'installer sur les nœuds deux interfaces de communication [38]. La première est dédiée à la transmission des données et consomme beaucoup d'énergie pour assurer cette fonction, relativement à la seconde qui supporte le système WuR et consomme très peu. Dans le modèle proposé, les nœuds désirant envoyer des paquets à leurs voisins vont d'abord envoyer un signal de réveil, puis transmettre les données lorsqu'ils sont en état de recevoir. Le gain énergétique vient du fait que l'interface de communication dédiée au système WuR, en mode idle, consomme beaucoup moins d'énergie que l'interface utilisée pour les données et le protocole réseau. Les auteurs montrent ainsi une économie d'énergie de plus de 50% par rapport aux solutions de Duty Cycle standards. Des méthodes n'utilisant qu'une seule interface ont également été proposées. Par exemple, il est considéré dans [39] que les interfaces réseau des nœuds peuvent être configurées à la volée pour fonctionner dans des états similaires aux interfaces WuR. Pour ce faire, elles doivent être équipées d'un circuit pouvant prendre en charge ce mode de fonctionnement à basse consommation.

Ces systèmes permettent donc d'obtenir de réels gains en termes d'économie d'énergie, tout en limitant les impacts négatifs sur les performances du réseau. Dans une comparaison entre une solution WuR applicable et un certain nombre de solutions Duty Cycle, les auteurs montrent que dans la majorité des scénarii étudiés, la solution WuR est comparable au Duty Cycle le moins efficace énergétiquement pour les résultats relatifs aux performances du réseau, mais surpasse toutes les autres solutions en termes énergétiques. Ils suggèrent alors une transition des systèmes d'économie d'énergie à base de Duty Cycle au WuR. Il faut cependant noter que les solutions WuR nécessitent un investissement matériel non négligeable et qu'elles sont par conséquent difficilement (voire pas du tout) applicables sur des réseaux existants.

2.4 Agrégation

Dans les réseaux, le terme agrégation peut être utilisé dans plusieurs situations. Par exemple, [40] propose une solution basée sur de l'agrégation de paquets afin d'améliorer les performances des applications de voix sur IP. L'agrégation de paquets consiste à rassembler en un seul paquet plusieurs - de plus petite taille - ayant la même destination. Cette technique permet de réduire la taille totale des en-têtes de paquets et ainsi

2.5 Conclusion

la proportion de temps passé en mode transmission. Cette méthode permet également de réduire le nombre de paquets transmis et le nombre de collisions (donc le besoin en retransmission), ce qui offre des opportunités d'économie d'énergie. L'agrégation de données est également une méthode classiquement utilisée dans les réseaux. Principalement développée dans les réseaux de capteurs [41], elle traite les données au fur et à mesure qu'elles sont relayées par les capteurs vers le point de collecte. L'objectif consiste à combiner les données entre elles (par de la suppression de redondance, un calcul de max/min, etc.) pour réduire le nombre de transmissions et donc la consommation énergétique du réseau. Enfin, il existe aussi l'agrégation de flux, utilisée notamment dans [42], permettant de collecter des données de télémétrie dans un réseau de manière optimisée en termes de performances, de surveillance ou encore de sécurité.

Dans cette thèse, nous appelons agrégation du flux le mécanisme consistant à rassembler plusieurs flux sur un même chemin. Étant donné qu'un nœud consomme principalement lorsqu'il est dans un état d'attente [2], les solutions d'économie d'énergie se basant sur l'extinction des nœuds s'avèrent très pertinentes. La solution proposée dans [43] utilise la programmation linéaire, un outil d'optimisation mathématique, afin de maximiser le nombre de nœuds pouvant être éteints dans un réseau sans fil multi-sauts en fonction du trafic. Les auteurs modélisent donc un problème de routage de flux sous contraintes linéaires en nombres entiers pouvant ensuite être résolu par des méthodes qui seront présentées dans le chapitre 3. Toutefois, ce travail s'adresse à des réseaux d'opérateurs dont le rôle applicatif est de fournir un accès internet à leurs utilisateurs. Cette hypothèse suggère donc l'existence de destinataires fixes (les passerelles fournissant la connexion), ainsi que la capacité pour le gestionnaire du réseau de les contrôler de manière centralisée. La solution n'est donc pas appropriée pour des réseaux plus dynamiques, en termes de flux ou de nœuds, du fait du nombre de messages de mise à jour qui devraient être envoyés au contrôleur central. De plus, l'optimum est obtenu en exécutant la solution proposée de façon itérative et progressive vis-à-vis du nombre de nœuds à éteindre. Cette approche n'est donc pas efficace lorsqu'il s'agit de fournir une route avec un faible délai.

2.5 Conclusion

L'aspect énergétique dans les réseaux sans fil multi-sauts a été longuement étudié. L'utilisation de métriques basées sur l'énergie a permis d'étendre les algorithmes de routage classiques afin d'améliorer les performances des réseaux et leur consommation énergétique. Malheureusement, la grande majorité des travaux existants visent à amé-

liorer l'efficacité énergétique des réseaux de capteurs, fortement contraints du point de vue énergétique, ou des réseaux mobiles. Dans les deux cas, les auteurs considèrent que les nœuds sont alimentés par des batteries. L'objectif de ces travaux est donc de maximiser la durée de vie du réseau. Cette notion de "durée de vie" est souvent redéfinie [44] pour répondre aux objectifs que se fixent les auteurs, ce qui rend la comparaison des solutions difficilement pertinente. De même, les techniques d'agrégation voient leur définitions adaptées selon le type de réseau dans lequel elles sont appliquées.

Le travail présenté dans cette thèse s'adresse principalement aux réseaux sans fil multi-sauts ayant des contraintes énergétiques plus souples. Dans les chapitres suivants, nous considérons la consommation énergétique globale du réseau et cherchons à la minimiser. L'objectif est bien de réduire l'empreinte énergétique du réseau dans son ensemble. Pour y parvenir, nous proposons des solutions (centralisées puis distribuées) minimisant le nombre de nœuds actifs [43]. Notre approche se base sur le routage et consiste à rassembler les flux du réseau sur un minimum de chemins afin de maximiser le nombre de nœuds pouvant être éteints, tout en garantissant la qualité de service des flux en termes de débits demandés. Ceci va à contre sens du consensus établi en matière d'efficacité énergétique dans les réseaux sans fil multi-sauts et des méthodes existantes dans cette partie. Le tableau suivant résume les différentes solutions présentées et notre proposition d'agrégation de flux, en fonction de critères de consommation énergétique, de qualité de service et d'applicabilité.

Algorithmes	Durée de vie	Energie globale	QoS	Interférence	Applicable
Routage réactif	Oui	Non Considéré	Parfois	Non	Oui
Routage proactif	Oui	Non Considéré	Parfois	Non	Oui
Duty Cycle	Oui	Non Considéré	Non	Non	Oui
Wake up Radio	Oui	Non Considéré	Non	Non	À étudier
Contrôle de topologie	Oui	Non Considéré	Parfois	Parfois	Parfois
Agrégation de flux	À étudier	Oui	Oui	Oui	Oui

FIGURE 2.1 – Méthodes existantes et positionnement de notre solution.

2.5 Conclusion

Chapitre 3

Agrégation de flux

Le problème que nous cherchons à résoudre est un problème de routage de flux avec minimisation de la consommation énergétique globale d'un réseau pour un ensemble de flux donné. Il s'agit de déterminer dans un réseau de communication les chemins à emprunter afin de permettre le transfert de données entre les sources et les destinations considérées. Ce type de problème peut être résolu efficacement et de façon optimale grâce à des méthodes d'optimisation sous contraintes linéaires et une modélisation sous forme de programme linéaire en nombres entiers (cas particulier de la programmation mathématique). Cette modélisation, triviale pour des problèmes de routage simples, doit toutefois prendre en compte un modèle de consommation énergétique permettant d'atteindre notre objectif de minimisation de la consommation globale du réseau.

Le modèle d'énergie à états présenté dans la Section 2.3 s'avère justement adapté à cette considération et simple à mettre en œuvre car n'utilisant qu'une seule variable pour déterminer la consommation énergétique globale d'un nœud : le temps passé par celui-ci dans les différents états de fonctionnement. Dans ce chapitre, nous établissons le lien entre la minimisation de la consommation énergétique d'un réseau sans fil multi-sauts et l'agrégation de flux telle que nous la considérons (c.f. Section 2.4), puis reprenons l'approche développée dans [43] en l'améliorant avec une solution exacte, directe et plus générique. Cette dernière consistait à router un ensemble de flux dans un réseau mesh (où les destinations des flux sont les passerelles avec l'Internet) de manière à maximiser le nombre de nœuds pouvant être éteints.

Pour pouvoir exprimer notre problème sous forme mathématique, un modèle capable de représenter un réseau sans fil multi-sauts et ses propriétés doit être utilisé. La plupart des travaux modélisent ce type de réseau par un graphe orienté où les nœuds représentent les dispositifs (routeurs, capteurs, antennes, smartphones, etc.) compo-

sant le réseau et les liens la capacité pour deux nœuds adjacents de communiquer. À ce graphe, qui ne représente que la topologie du réseau, viennent s'ajouter un ensemble modélisant les flux de données présents dans le réseau et un ensemble de zones d'interférences (généralement appelées cliques) servant à modéliser une contrainte physique propre au réseau sans fil et influençant la capacité du réseau.

Le développement d'une solution de routage optimale minimisant la consommation énergétique globale du réseau est une première étape permettant de servir de référence (ou benchmark) et de vérifier si les objectifs visés sont atteints. En effet, il n'est pas question d'avoir dans le réseau un nœud particulier dont le rôle serait de rassembler toutes les informations relatives à l'état du réseau (nœuds, liens, flux, interférences, etc.), puis calculer les routes que les flux doivent suivre pour atteindre leur destination, et enfin diffuser ces routes à l'ensemble du réseau. Comme nous l'avons vu dans le chapitre précédent, une telle méthode ne serait pas adaptée aux réseaux sans fil multi-sauts. C'est pourquoi nous proposerons dans le chapitre suivant une approche distribuée, dont les résultats seront très proches de ceux obtenus avec la solution optimale.

Dans les prochaines sections, nous décrivons donc une solution de routage optimale visant à minimiser le nombre de nœuds utilisés. Le modèle énergétique à états met en évidence le fait qu'un nœud consomme beaucoup d'énergie lorsqu'il est dans un état actif, mais aussi que la différence de consommation entre les différents états actifs (Tx, Rx et Idle) n'est pas significative comparée à celle qu'il y a avec l'état de veille [2]. Autrement dit, la consommation énergétique d'un nœud ne varie pas significativement en fonction de sa charge de travail lorsque celui-ci est actif. Par conséquent, il est plus intéressant de voir un nœud ayant une faible charge de travail s'éteindre - ou se mettre en veille - en redistribuant cette charge sur ses voisins que de maintenir ce nœud actif, qui consomme alors beaucoup d'énergie inutilement (c'est-à-dire pour une contribution minime vis-à-vis du réseau).

Bien entendu, cette approche n'est pas sans risque, le principal étant qu'en concentrant les flux sur un nombre réduit de nœuds, nous réduisons par la même occasion la capacité du réseau à transmettre des données. Bien que la minimisation du nombre de nœuds actifs réduise l'interférence et donc les retransmissions (limitant naturellement l'impact de l'agrégation de flux), un mécanisme de sécurité doit être mis en place afin de garantir un certain niveau de qualité de service, exprimé dans cette thèse en termes de débit (les débits demandés par les flux doivent être respectés).

Les conséquences de l'agrégation de flux peuvent toutefois être réduites grâce à l'utilisation de techniques de codage réseau [45, 46]. En effet, l'agrégation de flux, en rassemblant les flux sur un nombre minimum de nœuds, maximise les opportunités pour réaliser du codage réseau inter-flux. Cela permet, entre autres, de réduire le nombre global de transmissions nécessaires et donc la charge du réseau. Cette technique peut également être utilisée comme un moyen d'améliorer davantage l'agrégation de flux. En réduisant la charge du réseau au niveau des nœuds agrégeant les flux, il est possible d'en agréger de nouveaux qui, sans codage, ne pouvaient l'être. Il existe donc une véritable synergie entre les deux mécanismes.

La première partie de ce chapitre abordera en détail le modèle utilisé pour représenter le réseau et les interférences. La seconde partie présentera la programmation linéaire en nombres entiers : la méthode, la résolution et son application dans un problème de réseau. Puis nous développerons une solution de routage d'un ensemble de flux minimisant la consommation énergétique globale du réseau. La partie 3.4 décrira notre approche simplifiant ce problème en la minimisation du nombre de nœuds actifs. Nous détaillerons ensuite le cas de l'utilisation du codage réseau afin d'améliorer davantage l'agrégation de flux et/ou de limiter l'impact de cette dernière sur la capacité du réseau. Enfin, une conclusion résumera les résultats établis dans ce chapitre, rappellera les limites de la solution proposée et abordera la méthode retenue pour y remédier. Cette dernière fera l'objet du chapitre suivant et ouvrira la voie vers une intégration de l'agrégation de flux dans des algorithmes de routage distribués.

3.1 Notations et modèle de réseau étudié

3.1.1 Topologie de réseau

Un réseau sans fil multi-sauts est généralement modélisé sous la forme d'un graphe orienté $G = (V, E)$, où V est l'ensemble des nœuds composant le réseau et E l'ensemble des liens entre les nœuds. Un couple (i, j) , avec i et j deux nœuds quelconques du réseau, appartient à E si et seulement s'il existe un lien de communication de i vers j . Par exemple, la Figure 3.1 représente une topologie en ligne de cinq nœuds. Dans cette thèse, nous considérons qu'un nœud peut transmettre des données à un autre si ce dernier se trouve dans la zone de transmission du premier. Enfin, pour des raisons de clarté, nous considérons que les nœuds bénéficient de la même distance de transmission, donc si un couple (i, j) appartient à E , alors le couple (j, i) en fait également

3.1 Notations et modèle de réseau étudié

partie (cette hypothèse peut facilement être levée). Par ailleurs, un lien (i, j) dispose d'une capacité $C_{i,j}$, représentant le débit maximal avec lequel des données peuvent être transmises sur ce lien, et d'un poids $W_{i,j}$ représentant le coût d'utilisation de ce lien pour la transmission de données. Un flux f est représenté par un couple $((S_f, D_f), d_f)$, où S_f et D_f sont respectivement la source et la destination du flux et d_f la qualité de service demandée (en termes de débit) du flux.

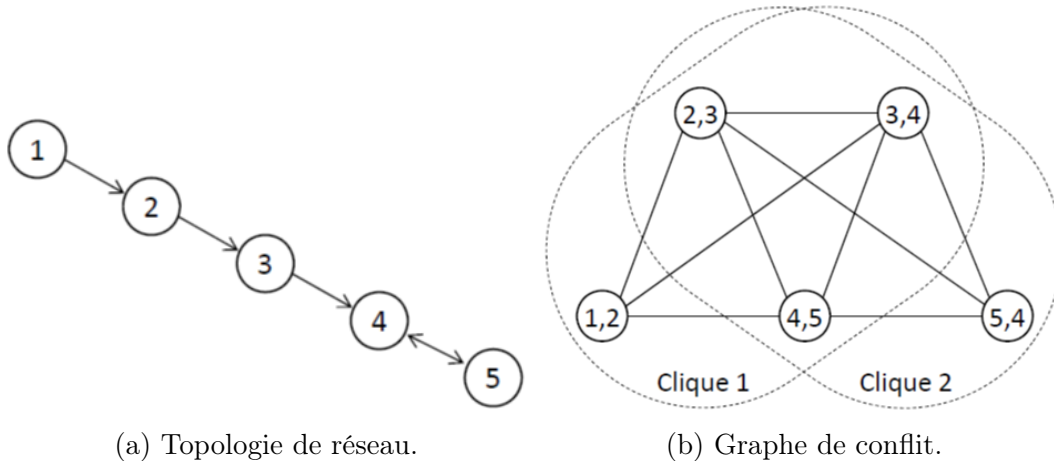


FIGURE 3.1 – Exemple de réseau sans fil multi-sauts et son graphe de conflit associé.

3.1.2 Modèle d'interférences

Le phénomène d'interférences est l'un des principaux problèmes dans les travaux relatifs aux réseaux sans fil multi-sauts. Il peut être modélisé de deux manières [47]. La première considère les interférences comme du bruit. En prenant en compte la puissance de transmission des nœuds et la position de tous les nœuds transmettant des données au même moment, il est possible de calculer un ratio signal sur bruit pour chacun des paquets reçus par un nœud. Si ce ratio est supérieur à un seuil fixé, le paquet est considéré comme correctement reçu. Dans le cas contraire, il est supposé qu'une erreur est survenue lors de la transmission et donc que le paquet n'est pas reçu. La deuxième méthode consiste à considérer que la transmission d'un paquet entre deux nœuds i et j s'est correctement déroulée si la distance séparant i et j est inférieure ou égale à la distance de transmission de i et s'il n'y a pas d'autre nœud dans leur voisinage qui transmette simultanément.

Par la suite, nous considérerons la seconde approche afin de modéliser les interférences. En effet, l'un des principes fondamentaux dans le développement des protocoles réseaux est la séparation des couches (ou niveaux). Puisque ce travail concerne le niveau IP (routage), il est préférable d'éviter de recourir à des informations de niveaux

différents (par exemple la puissance du signal est une information liée à la couche physique). De plus, ce modèle d'interférence permet de construire un graphe de conflit représentant les relations d'interférence entre les liens du réseau [48]. De ce graphe, il est possible de déduire une formule validant ou non la possibilité d'insérer un ensemble de flux dans le réseau. Les liens interfèrent entre eux selon un modèle d'interférence à N -sauts. Nous considérerons $N = 2$, valeur classiquement admise dans l'état de l'art. Ainsi, un nœud interfère avec ses voisins et les leurs. Par conséquent, pour deux nœuds interférants, nous supposons que tous les liens sortants de l'un interfèrent avec tous les liens adjacents de l'autre et réciproquement (les liens entrants de ces deux nœuds peuvent ne pas interférer entre eux selon la situation). En calculant l'ensemble des cliques (sous-graphes fortement connexes maximaux), noté C , du graphe de conflit résultant de ces liens interférants, nous obtenons les ensembles des liens ne pouvant pas être utilisés simultanément dans le réseau sous peine de provoquer des erreurs de transmission. Par exemple, avec un modèle d'interférence à 2 sauts, le graphe de conflit de la topologie en ligne illustrée ci-dessus est celui de la Figure 3.1.

Ces cliques nous permettent ensuite de déduire la contrainte suivante de capacité avec prise en compte des interférences, ou contrainte de cliques [49] :

$$\forall c \in C, \sum_{(i,j) \in c} \frac{d_{i,j}}{C_{i,j}} \leq 1$$

Cette contrainte calcule le taux d'utilisation de chaque clique comme la somme des taux d'utilisation des liens la composant. Ici, $d_{i,j}$ représente la capacité totale réservée sur le lien (i, j) et dépend de la manière dont un ensemble de flux est routé sur le réseau. Nous considérons qu'un réseau n'est pas surchargé et donc que cet ensemble de flux est réalisable, lorsque le taux d'utilisation de chaque clique est inférieur à 100%.

3.2 Programmation linéaire en nombres entiers

3.2.1 Présentation

Depuis de nombreuses années, la programmation mathématique [50, 51] s'est avérée être un outil précieux pour modéliser et résoudre de nombreux problèmes du monde réel dans des domaines variés. Celui des réseaux informatiques voit cette méthode appliquée pour résoudre des problèmes de dimensionnement, d'amélioration de performance ou de routage [52, 53, 54]. Une modélisation (formulation mathématique) du problème traité dans cette thèse s'avère être non linéaire en variables mixtes. Toutefois, des méthodes de linéarisations classiques des contraintes (logiques) et de la fonction objectif peuvent

3.2 Programmation linéaire en nombres entiers

être utilisées afin de se ramener à une formulation linéaire. L'avantage de cette dernière est qu'il existe aujourd'hui des solveurs d'optimisation linéaire, tel que CPLEX [55], très efficaces en termes de temps CPU et ce, pour des problèmes de grandes tailles.

3.2.2 Un exemple

Nous proposons ici un exemple de problème académique d'optimisation combinatoire modélisé sous la forme d'un programme linéaire en nombres entiers afin de faciliter la lecture et la compréhension des modèles proposés dans cette thèse. Le problème du sac à dos est un cas classique de problème d'optimisation combinatoire, modélisé sous la forme d'un programme linéaire en nombres entiers (ou 0-1). Ce problème consiste à remplir un sac, de capacité limitée, avec différents objets de façon à maximiser la valeur totale du sac en fonction du poids et de la valeur de chaque objet disponible. Ainsi, supposons que nous ayons un sac d'une capacité de 10 kg et un ensemble de quatre objets $O = \{1, 2, 3, 4\}$ de valeurs respectives $V = \{4, 8, 10, 3\}$ et ayant des poids respectifs de $C = \{1, 3, 8, 2\}$. Quels sont, parmi les quatre objets disponibles, ceux à sélectionner et à placer dans le sac afin de maximiser la valeur totale dans le sac ? Ce problème se modélise de la façon suivante :

$$(1) \text{ Max } z = 4x_1 + 8x_2 + 10x_3 + 3x_4$$

Soumis à :

$$(2) x_1 + 3x_2 + 8x_3 + 2x_4 \leq 10$$

$$(3) x_i \in \{0, 1\}, i \in O$$

Dans cette formulation, la ligne (1) représente la fonction objectif à maximiser et traduit la valeur totale du sac. Les éléments $\{x_1, x_2, x_3, x_4\}$ sont les variables de décision que nous cherchons à déterminer et prenant leurs valeurs dans l'ensemble discret $\{0,1\}$, comme le précise la contrainte (3). Ces variables traduisent le fait que l'objet qu'elles représentent est placé dans le sac ou non. Plus précisément, si la variable $x_i = 1$, alors l'objet i est mis dedans ; si $x_i = 0$, il ne doit pas y être. Enfin, la contrainte (2) représente le niveau de remplissage du sac en fonction des poids des objets et assure que la contenance du sac n'est jamais dépassée (on ne peut pas mettre plus d'objets dans le sac qu'il ne peut en contenir). Cette contrainte est généralement appelée contrainte de sac à dos (ou encore de capacité).

Les programmes linéaires en nombres entiers de ce type sont connus pour être NP difficiles [56], c'est-à-dire qu'il n'existe probablement pas d'algorithmes pour les résoudre en temps polynomial. Toutefois, des algorithmes exacts de type séparation et

évaluation (Branch & Bound [57]) sont fréquemment utilisés. Ce sont des méthodes par énumérations implicites se basant sur la création d'un arbre de recherche des solutions. Chaque nœud construit représente une partie d'une solution existante, autrement dit une partie des variables ont été fixées à des valeurs entières, ainsi que la valeur potentiellement optimale Z_{cont} de la fonction objectif. Cela permet de déterminer la nécessité de développer un nœud si une solution entière Z a déjà été trouvée, puisque la valeur de Z_{cont} représente la solution optimale de ce que nous appelons la relaxation continue du problème en nombres entiers. Cette relaxation résout donc le problème en considérant les variables comme des variables continues tout en fixant à des valeurs entières les variables de la solution partielle étudiée. En effet, l'inclusion de l'ensemble des solutions entières dans celui des solutions continues amène la relation suivante : la valeur de la fonction objectif à l'optimum d'un problème linéaire en nombres entiers sera toujours inférieure ou égale à la valeur à l'optimum de sa relaxation continue dans un contexte de maximisation. Par conséquent, si au moment d'étudier un nœud de l'arbre, Z_{cont} est inférieure à la valeur de la solution entière (toutes les variables ont une valeur entière) précédemment trouvée, alors il n'est pas nécessaire de développer davantage ce nœud. Il en est de même si la relaxation continue du problème n'a pas de solution.

La figure 3.2 montre un exemple de résolution du programme de sac à dos avec la méthode du Branch & Bound. X représente la solution partielle qui est évaluée sur un nœud de l'arbre. Z_{max} représente la valeur de la dernière meilleure solution entière trouvée. Chaque nœud de l'arbre dispose d'un numéro représentant l'ordre d'évaluation (calcul de la relaxation continue) du nœud. Le développement d'un nœud (apparition des nœuds fils) représente l'attribution des différentes valeurs entières possibles à l'une des variables de décision du problème. Ainsi, une solution entière au problème est trouvée lorsque X contient toutes les variables de décision (une feuille de l'arbre est atteinte). Enfin, une heuristique doit être utilisée pour déterminer quel nœud développer et quelle variable utiliser pour le développement. Ces deux paramètres font l'objet de nombreuses recherches [58] afin d'améliorer l'exécution de cet algorithme et ainsi optimiser davantage les solveurs existants. Pour cet exemple et par souci de simplification, nous décidons de développer le nœud proposant la plus grande relaxation continue (Z_{cont}) et fixer les variables en suivant l'ordre de leur indice.

Il faut noter que ceci n'est qu'un exemple de programme mathématique en nombres entiers et de l'algorithme pour le résoudre à l'optimum. La partie suivante présente la modélisation d'un problème de routage classique dans les réseaux et donne les contraintes de base de tous les programmes de routage qui seront définis par la suite avec cette méthode de modélisation.

3.2 Programmation linéaire en nombres entiers

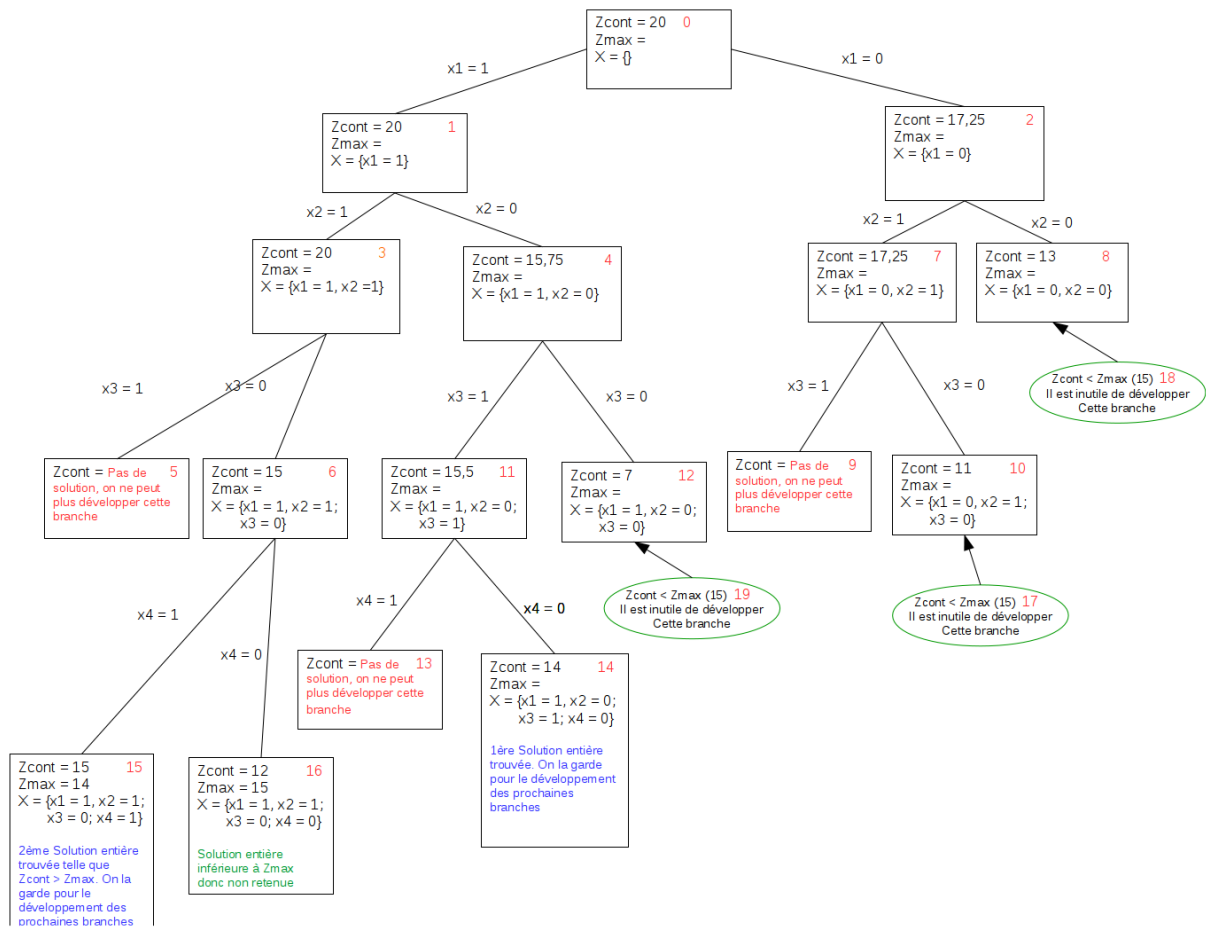


FIGURE 3.2 – Algorithme Branch&Bound

3.2.3 Modélisation d'un problème de routage - Cas du plus court chemin (ILP)

Comme pour le problème de sac à dos, celui du plus court chemin est un problème de routage classique que l'on peut trouver dans les réseaux de communication et qui peut être résolu par la programmation mathématique [59]. Il consiste à chercher dans un réseau, composé de plusieurs nœuds communicants, la chaîne de nœuds ou de liens selon les modèles utilisés, permettant aux deux extrémités d'échanger des messages par retransmissions successives le long de cette chaîne. En réseau, il s'agit d'un flux de données. Dans cette thèse, il sera considéré dans la modélisation un ensemble de flux unidirectionnels, noté F .

Ainsi, le problème de plus court chemin peut être modélisé au moyen du programme linéaire en nombres entiers, ou ILP (Integer Linear Programming), suivant :

$\text{Min } z = \sum_{f \in F} l_f$ <p>s.c.</p> <p>(1) $d_{i,j} < C_{i,j}, \forall (i, j) \in E$</p> <p>(2) $\sum_{(i,j) \in E} x_{i,j}^f - \sum_{(j,k) \in E} x_{j,k}^f = 0, \forall f \in F, \forall j \in V - \{S_f, D_f\}$</p> <p>(3) $\sum_{(i,S_f) \in E} x_{i,S_f}^f = 0, \forall f \in F$</p> <p>(4) $\sum_{(S_f,i) \in E} x_{S_f,i}^f = 1, \forall f \in F$</p> <p>(5) $\sum_{(D_f,i) \in E} x_{D_f,i}^f = 0, \forall f \in F$</p> <p>(6) $\sum_{(i,D_f) \in E} x_{i,D_f}^f = 1, \forall f \in F$</p> <p>(7) $\sum_{f \in F} x_{i,j}^f \cdot d_f = d_{i,j}, \forall (i, j) \in E$</p> <p>(8) $\sum_{(i,j) \in E} x_{i,j}^f \leq 1, \forall f \in F, \forall i \in V - \{D_f\}$</p> <p>(9) $\sum_{(i,j) \in E} x_{i,j}^f \cdot W_{i,j} = l_f, \forall f \in F$</p> <p>$x_{i,j}^f \in \{0, 1\}, \forall (i, j) \in E, \forall f \in F$</p> <p>$d_{i,j}^f \in \mathbb{R}^+, \forall (i, j) \in E, \forall f \in F$</p> <p>$l_{i,j}^f \in \mathbb{R}^+, \forall (i, j) \in E, \forall f \in F$</p>

Dans ce programme, les variables de décision sont $x_{i,j}^f$, $d_{i,j}$ et l^f . Elles représentent respectivement le fait qu'un flux soit routé sur le lien (i, j) , la capacité réservée sur ce lien (i, j) et la longueur du chemin attribué au flux f . Il faut toutefois noter que ces variables sont fortement dépendantes entre elles et que $d_{i,j}$ et l^f peuvent être déduites de $x_{i,j}^f$. La fonction objectif qui est à minimiser dans ce programme représente la somme des longueurs des chemins attribués aux flux coexistant dans le réseau. La minimisation de cette fonction revient donc à chercher la solution au problème de plus court chemin. Les contraintes du programme linéaire définissent les propriétés du routage et sont détaillées ci-dessous (les contraintes (2) à (6) sont classiques dans ce type de problème) :

- (1) Assure que la capacité réservée sur un lien ne dépasse pas la capacité totale de ce lien (contrainte de capacité). Dans le cadre d'une application aux réseaux sans fil multi-sauts, cette contrainte doit prendre en considération les interférences (c.f. Section 3.1.2). La solution ILP devient alors I2ILP (Intra-flow Interference-aware ILP) [60].
- (2) Modélise la loi de conservation des flots. Elle assure que pour chaque nœud intermédiaire (ni source ni destination) un flux passant par des liens entrants quitte le nœud par autant de liens sortants.
- (3) Permet d'éviter qu'un flux ne boucle sur sa propre source.
- (4) Active une source en forçant l'un de ses liens sortants à router le flux dont il est source.
- (5) Assure qu'un flux, après être arrivé à destination, n'en sortira pas.

3.3 Minimisation de la consommation énergétique

- (6) Active la destination d'un flux en assurant qu'un de ses liens est utilisé pour router le flux qu'il doit recevoir.
- (7) Calcule la capacité réservée sur un lien en faisant la somme des demandes des flux passant par ce lien.
- (8) Assure le routage mono-chemin, en garantissant qu'un nœud intermédiaire ne route un flux que sur un seul de ses liens sortants.
- (9) Calcule la longueur des chemins trouvés en sommant le poids des liens par lesquels passent les flux.

La dimension de ce programme d'optimisation s'avère bien supérieure à celle du problème du sac à dos présenté dans la partie précédente, tant en nombre de variables ($O(|F|.|E| + |F| + |E|)$) qu'en nombre de contraintes ($O(|F| + |E| + |V| + |F|.|V|)$). De plus, il s'agit ici d'une complexité minimale pour un problème de routage classique. Plus nous introduirons de contraintes pour ajouter de nouvelles fonctionnalités à ce programme, plus la complexité augmentera. La dimension des programmes produits est donc telle qu'il deviennent difficiles (dans un temps raisonnable) à résoudre de façon exacte juste en utilisant des solveurs classiques [55]. Le développement d'algorithmes de résolution dédiés à ce problème serait plus adéquat.

La recherche de telles méthodes de résolution de programmes d'optimisation linéaires en nombres entiers est hors du champ de cette thèse, qui se concentre sur la modélisation des problèmes en réseaux. Nous utiliserons les solveurs existants pour obtenir la solution optimale à ces problèmes. Et bien que ces outils informatiques ne soient pas optimisés pour résoudre les problèmes qui nous concernent, ils restent toutefois des outils génériques capables de répondre efficacement et de manière exacte pour des instances de taille raisonnable.

3.3 Minimisation de la consommation énergétique

3.3.1 Modèle énergétique proposé

Dans cette thèse, nous cherchons à réduire la consommation énergétique globale des réseaux sans fil multi-sauts, c'est-à-dire l'énergie consommée par l'ensemble des nœuds du réseau sur une unité de temps donnée. Pour ce faire, nous utilisons dans un premier temps un modèle énergétique à états tel que défini dans la Section 2.3. De plus, nous supposons que, par souci d'économie d'énergie, les nœuds transmettent systématiquement au débit maximal permis par les limites physiques du lien (débit de

transmission) afin de minimiser le temps passé dans l'état de transmission [2]. Cette hypothèse nous permet de déduire le temps passé par un nœud dans chaque état en fonction du taux d'utilisation de ses liens entrants (pour l'état de réception) et sortants (pour l'état de transmission). En effet, sachant qu'un nœud transmet ou reçoit toujours au maximum de la capacité du lien correspondant ($C_{i,j}$), le temps passé en mode transmission ou réception pour les données d'un flux ayant une demande d_f est égal à $\frac{d_f}{C_{ij}}$. Connaissant le temps passé dans chacun de ces deux états, il est possible d'en déduire le temps pendant lequel le nœud est dans un état passif (attente ou veille).

Nous considérons les comportements des nœuds suivants pendant l'unité de temps étudiée :

- Un nœud qui est sollicité pour transmettre et/ou recevoir des données est en mode attente (Idle) tout le reste du temps.
- Un nœud qui n'a rien à transmettre ou à recevoir (il ne fait partie d'aucune route sélectionnée) est dans l'état de veille (Sleep). Ceci représente le fait qu'un nœud non utile dans le réseau s'éteint pour réduire sa consommation énergétique.

Ainsi, la consommation énergétique globale d'un réseau se modélise de la façon suivante :

$$\sum_{i \in V} E_i$$

s.c :

- (1) $E_i = E_i^T + E_i^R + E_i^P, \forall i \in V$
- (2) $E_i^T = E^T \cdot (\sum_{(i,j) \in E} \frac{d_{i,j}}{C_{i,j}}), \forall i \in V$
- (3) $E_i^R = E^R \cdot (\sum_{(j,i) \in E} \frac{d_{j,i}}{C_{j,i}}), \forall i \in V$
- (4) $E_i^I = E^I \cdot (1 - \sum_{(i,j) \in E} \frac{d_{i,j}}{C_{i,j}} - \sum_{(j,i) \in E} \frac{d_{j,i}}{C_{j,i}}), \forall i \in V$
- (5) $E_i^S = E^S \cdot (1 - \sum_{(i,j) \in E} \frac{d_{i,j}}{C_{i,j}} - \sum_{(j,i) \in E} \frac{d_{j,i}}{C_{j,i}}), \forall i \in V$
- (6.1) $E_i^S - E_i^P - E^T \cdot x_i \leq 0, \forall i \in V$
- (6.2) $E_i^I - E_i^P - E^T \cdot (1 - x_i) \leq 0, \forall i \in V$

$x_i \in \{0, 1\}, \forall i \in V$

$(E_i, E_i^T, E_i^R, E_i^I, E_i^S, E_i^P) \in \mathbb{R}^{+6}, \forall i \in V$

3.3 Minimisation de la consommation énergétique

Dans cette formulation, les variables de décision sont $d_{i,j}$, x_i , E_i , E_i^T , E_i^R , E_i^S , E_i^I et E_i^P . La variable $d_{i,j}$ est définie dans le chapitre précédent et représente la capacité réservée sur un lien (i, j) . Le fait qu'un nœud i soit utilisé pour router des flux est représenté par la variable booléenne x_i et sa consommation totale par E_i . Les variables E_i^T , E_i^R , E_i^I et E_i^S définissent l'énergie consommée par le nœud i lorsque celui-ci est dans l'état Tx, Rx, Idle ou Sleep, respectivement. Lorsque le nœud est passif, la variable E_i^P donne sa consommation énergétique, quelque soit son mode (en attente ou en veille). Quant aux constantes E^T , E^R , E^I et E^S , elles représentent la consommation d'un nœud par unité de temps en fonction de l'état dans lequel il se trouve (respectivement en transmission, réception, attente et veille).

Ce modèle définit donc la fonction de consommation énergétique du réseau ainsi que les contraintes qui en déterminent les composantes :

- La contrainte (1) calcule la consommation totale E_i de chaque nœud i .
- Les contraintes (2) et (3) définissent l'énergie utilisée par les nœuds lorsqu'ils sont dans un état actif (transmission ou réception).
- Les contraintes (4) et (5) définissent l'énergie consommée par les nœuds lorsqu'ils sont inactifs (attente ou veille).
- Les contraintes (6.1) et (6.2) permettent de sélectionner la valeur (E_i^I ou E_i^S) qui doit être utilisée pour calculer l'énergie consommée par les nœuds lorsqu'ils sont inactifs. Lorsque $x_i = 1$, c'est-à-dire lorsque le nœud i est utilisé pour router un flux, la contrainte (6.1) devient moins restrictive que la contrainte (6.2), puisque la première implique la relation d'ordre $E_i^S - E^T < E_i^P$ et la seconde $E_i^I < E_i^P$. Or E^T est, par définition, l'énergie maximale consommée par les nœuds par unité de temps, à savoir l'énergie utilisée dans l'état de transmission. Donc $E_i^S - E^T < 0$, rendant la contrainte (6.1) toujours vraie. Ce n'est pas le cas de la contrainte (6.2) qui limite la valeur minimale de E_i^P par E_i^I . Dans le contexte d'une minimisation de E_i , E_i^P prend la valeur minimale qui lui est possible d'avoir, soit E_i^I .

Sur le même principe, lorsque $x_i = 0$, alors la contrainte (6.2) se retrouve effective.

Le modèle énergétique étant maintenant établi, nous pouvons aborder son application dans un problème de routage mono-chemin de flux dans un réseau sans fil multi-sauts, avec pour objectif la minimisation de la consommation énergétique globale du réseau.

3.3.2 Programme d'optimisation - Minimisation de l'énergie (MinEnergy)

L'un des avantages de la programmation linéaire est l'extrême modularité des modèles qu'il est possible de développer. L'insertion, l'extension ou le remplacement de contraintes peut suffire à ajouter des fonctionnalités aux modèles. Ainsi, en incluant à la modélisation du routage sur le plus court chemin les contraintes vues précédemment, il est possible d'ajouter la mesure de la consommation énergétique des nœuds. De plus, en modifiant la fonction objectif du modèle, nous changeons le comportement du programme linéaire. En utilisant la fonction de la consommation globale du réseau à la place de la fonction de longueur totale des chemins, nous obtenons le programme de routage minimisant la consommation énergétique globale du réseau suivant :

$$\text{Min } \sum_{i \in V} E_i$$

Tel que :

- (1) $\sum_{(i,j) \in c} \frac{d_{i,j}}{C_{i,j}} \leq 1, \forall c \in C$
- (2) $\sum_{(i,j) \in E} x_{i,j}^f - \sum_{(j,k) \in E} x_{j,k}^f = 0, \forall f \in F, \forall j \in V - \{S_f, D_f\}$
- (3) $\sum_{(i,S_f) \in E} x_{i,S_f}^f = 0, \forall f \in F$
- (4) $\sum_{(S_f,i) \in E} x_{S_f,i}^f = 1, \forall f \in F$
- (5) $\sum_{(D_f,i) \in E} x_{D_f,i}^f = 0, \forall f \in F$
- (6) $\sum_{(i,D_f) \in E} x_{i,D_f}^f = 1, \forall f \in F$
- (7) $\sum_{f \in F} x_{i,j}^f \cdot d_f = d_{i,j}, \forall (i,j) \in E$
- (8) $\sum_{(i,j) \in E} x_{i,j}^f \leq 1, \forall f \in F, \forall i \in V - \{D_f\}$
- (9) $x_i = \begin{cases} 0 & \text{si } d_{i,j} = d_{j,i} = 0, \forall (i,j), (j,i) \in E \\ 1 & \text{sinon} \end{cases}, \forall i \in V$
- (10) $E_i = E_i^T + E_i^R + E_i^P, \forall i \in V$
- (11) $E_i^T = E^T \cdot (\sum_{(i,j) \in E} \frac{d_{i,j}}{C_{i,j}}), \forall i \in V$
- (12) $E_i^R = E^R \cdot (\sum_{(j,i) \in E} \frac{d_{j,i}}{C_{j,i}}), \forall i \in V$
- (13) $E_i^I = E^I \cdot (1 - \sum_{(i,j) \in E} \frac{d_{i,j}}{C_{i,j}} - \sum_{(j,i) \in E} \frac{d_{j,i}}{C_{j,i}}), \forall i \in V$
- (14) $E_i^S = E^S \cdot (1 - \sum_{(i,j) \in E} \frac{d_{i,j}}{C_{i,j}} - \sum_{(j,i) \in E} \frac{d_{j,i}}{C_{j,i}}), \forall i \in V$
- (15.1) $E_i^S - E_i^P - E^T \cdot x_i \leq 0, \forall i \in V$
- (15.2) $E_i^I - E_i^P - E^T \cdot (1 - x_i) \leq 0, \forall i \in V$

$$x_{i,j}^f \in \{0, 1\}, \forall (i,j) \in E, \forall f \in F$$

$$d_{i,j}^f \in \mathbb{R}^+, \forall (i,j) \in E, \forall f \in F$$

$$x_i \in \{0, 1\}, \forall i \in V, \forall i \in V$$

$$(E_i, E_i^T, E_i^R, E_i^I, E_i^S, E_i^P) \in \mathbb{R}^{+6}, \forall i \in V$$

3.3 Minimisation de la consommation énergétique

Ce programme d'optimisation est principalement composé des "blocs" de contraintes et des variables de décision présentés dans les parties précédentes. Ainsi, les contraintes (1) à (8) sont celles définissant le routage et que nous appellerons dorénavant *contraintes de routage*. Il est toutefois important de souligner que la contrainte de capacité (1) utilisée ici est celle prenant en compte les interférences (cf. Section 3.1.2) et que la contrainte définissant la longueur des chemins a été supprimée car devenue inutile. Les contraintes (10) à (15) sont les contraintes du modèle énergétique. Enfin, la contrainte (9) est une contrainte logique permettant de modéliser le fait qu'un nœud est utilisé si l'un de ses liens sortants ou entrants est utilisé pour transmettre recevoir des données. Bien que cette contrainte ne soit pas linéaire, elle peut être facilement linéarisée [61] de la manière suivante :

$$(9.1) \text{Max}_{(i,j) \in E} (C_{i,j}) \cdot x_i - \sum_{(i,j) \in E} d_{i,j} \geq 0, \forall i \in V$$

$$(9.2) \text{Max}_{(j,i) \in E} (C_{j,i}) \cdot x_i - \sum_{(j,i) \in E} d_{j,i} \geq 0, \forall i \in V$$

$$(9.3) \sum_{j \in V} (d_{i,j} + d_{j,i}) - \text{Min}_{f \in F} (d_f) \cdot x_i \geq 0, \forall i \in V$$

Les contraintes (9.1) et (9.2) permettent de fixer x_i à la valeur 1 si du trafic est réservé sur les liens impliquant le nœud i . A l'inverse, si aucun flux ne passe par ce nœud, alors la contrainte (9.3) fixe x_i à la valeur 0.

3.3.3 Simulation et résultats numériques

Cette partie présente les performances, en termes de consommation énergétique et de charge du réseau, de notre solution et les compare à celles d'un algorithme de routage de plus court chemin avec prise en compte des interférences. Dans un premier temps, nous décrivons les outils de simulation développés pour générer les résultats présentés par la suite. Puis, nous abordons les paramètres déterminant les propriétés des simulations générées tout au long de la thèse. Nous présentons également les mesures retenues pour l'évaluation des différentes solutions qui ont été développées pour réduire la consommation énergétique globale des réseaux sans fil multi-sauts. Enfin, les résultats propres au programme d'optimisation présenté dans la partie 3.3 sont détaillés, confirmant notre intuition de départ quant à l'approche retenue par la suite pour résoudre notre problème d'économie d'énergie globale du réseau.

Outils de simulation

Pour les besoins de cette thèse, des outils ont été développés en C++ et en Python afin de générer (aléatoirement ou non), manipuler, stocker, évaluer et visualiser

des graphes de flux. Une interface a également été implémentée afin d'écrire des programmes d'optimisation sous contraintes linéaires à partir d'un graphe, de les résoudre via un appel au solveur CPLEX [55] et d'appliquer une solution renvoyée par ce dernier sur le graphe.

La figure 3.3 montre l'un des outils développés durant cette thèse. Il permet de générer un graphe, de placer des flux aléatoires dessus et de router ces flux selon plusieurs algorithmes de routage. Le cadre en haut à gauche permet d'avoir une vue graphique du graphe ainsi créé et indique les chemins utilisés par les flux. Le cadre en haut à droite regroupe plusieurs onglets permettant d'afficher le graphe sous forme textuelle (texte et sérialisé), des valeurs de charge ou de consommation énergétique. Le cadre du bas comprend différents champs permettant de régler les paramètres de génération des graphes (densité, nombre de nœuds, structure aléatoire ou en grille, etc.) et des flux (nombre de flux, demandes de QoS, durée, algorithmes de routage). Enfin, la liste des flux coexistant dans le réseau actuel permet de mettre en valeur un flux en particulier et de visualiser le chemin qui lui a été attribué. Cet outil fut très utile lors des premières étapes de la thèse afin de vérifier la cohérence des graphes générés et des solutions proposées par la résolution des programmes linéaires. Il fut également un outil d'aide à la recherche de la métrique de routage qui sera présentée dans le chapitre 4.

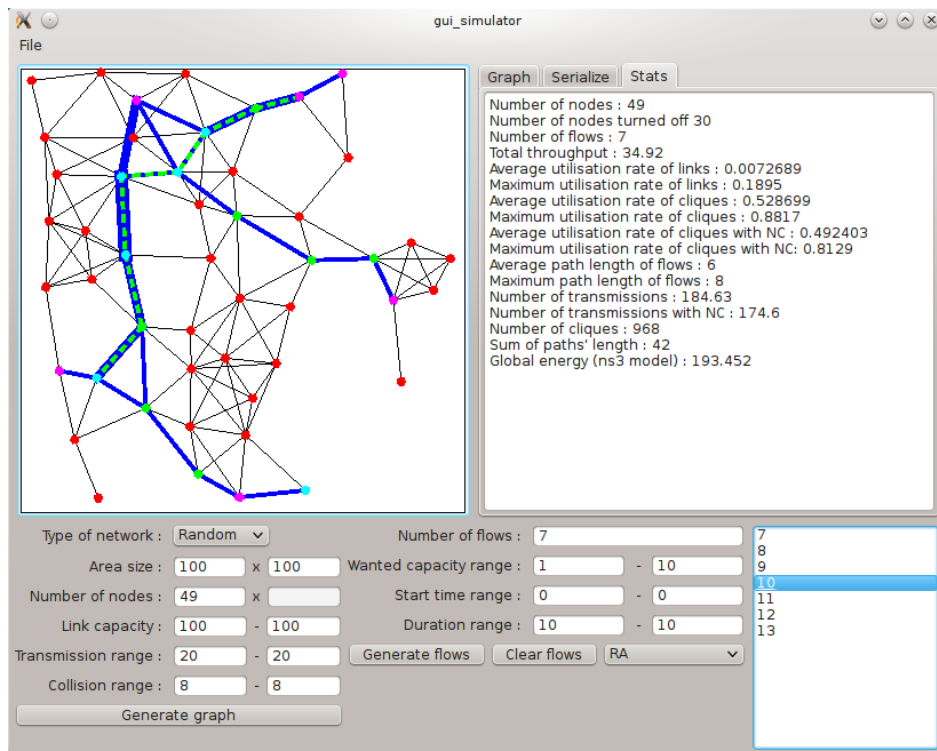


FIGURE 3.3 – Outil de génération et visualisation de graphe

Paramètres et scénarii de simulation

Les résultats présentés se basent sur des mesures moyennes (avec intervalle de confiance à 95%) obtenues après un grand nombre de simulations. Sauf exception (qui sera alors mentionnée), tous les résultats dans ce document sont produits sur un ensemble de graphes générés aléatoirement. Plus précisément, un graphe (topologie de réseau) est généré en plaçant les nœuds dans une zone de 100x100 unités de longueur. Chaque nœud dispose d'une distance de transmission de 20 unités et ne peut être placé à moins de 8 unités d'un autre nœud. Le graphe est généré de telle sorte qu'il soit toujours connexe et la capacité des liens est unique et fixée à 100 unités de capacité. Sur chaque graphe, un ensemble de flux est généré en sélectionnant aléatoirement pour chacun d'eux deux nœuds (source et destination) qui ne peuvent être voisins. La demande en capacité d'un flux est également sélectionnée aléatoirement, de façon uniforme, entre 1 et 10 unités et l'ensemble. Par ailleurs, l'ensemble de flux est toujours réalisable, c'est-à-dire qu'il existe au moins une route entre une source et sa destination telle que la contrainte de capacité avec interférence soit respectée. Ceci permet de garantir l'existence d'au moins une solution au problème d'optimisation linéaire. L'association d'un graphe G avec un ensemble de flux réalisable F sur ce graphe représente un scénario de simulation.

Les constantes de consommation énergétique retenues sont celles présentées dans [2] et sont les valeurs par défaut utilisées par le simulateur NS-3 [62], à savoir :

- $E^T = 1,140J$ (en mode Tx, état de transmission) ;
- $E^R = 0,940J$ (en mode Rx, état de réception) ;
- $E^I = 0.819J$ (en mode Idle, état d'attente) ;
- $E^S = 0.099J$ (en mode Sleep, état de veille). veille.

Ces valeurs ne prennent en compte que la consommation énergétique de l'interface de communication sans fil d'un nœud et non celle nécessaire à la mise sous tension de tous les composants du nœud.

Enfin, l'ensemble S des scénarii de simulation permet de faire varier le nombre de flux insérés dans le réseau et la topologie (en gardant toutefois la même densité). Cet ensemble est généré selon l'algorithme 1. Par conséquent, pour chaque simulation, 5000 scénarii sont testés, chacun représentant une topologie parmi cinq dans laquelle entre un et dix flux sont générés aléatoirement.

Algorithm 1 Génération des scénarii de simulation

```

1: procedure SCENARIOSGEN
2:    $S \leftarrow \{\}$  ▷ Initialise l'ensemble des scénarii.
3:   for  $i \leftarrow 1, 5$  do
4:      $G \leftarrow \text{new Graph}()$  ▷ Crée une nouvelle topologie.
5:     for  $j \leftarrow 1, 10$  do
6:       for  $k \leftarrow 1, 100$  do
7:          $S \leftarrow S \cup \{\text{new Scenario}(G, j)\}$  ▷ Crée un scénario en insérant  $j$  flux à  $G$ .
8:       end for
9:     end for
10:  end for
11:  return  $S$  ▷ Renvoie l'ensemble des scénarii de simulation.
12: end procedure

```

Mesures

Dans cette thèse, nous nous concentrons sur la consommation énergétique globale du réseau pour mesurer l'efficacité énergétique de nos algorithmes. La plupart des algorithmes de routage traitant le problème de l'économie d'énergie tendent à répartir de façon uniforme le trafic sur l'ensemble des nœuds du réseau, obligeant ainsi un maximum de nœuds à rester actifs sur la même période de temps. Si cette approche peut éventuellement être intéressante pour améliorer la capacité des réseaux sans fil multi-sauts, elle impose un surcoût énergétique fixe dû à la mise sous tension d'un nombre plus important de nœuds. Ce coût est lié à l'énergie nécessaire pour alimenter les circuits de base du nœud ou de son interface de communication sans fil. Par conséquent, bien que répartir uniformément la charge du réseau pour limiter l'impact de la consommation énergétique variable (dépendant du niveau d'activité) sur les nœuds soit efficace, cela implique aussi une consommation énergétique globalement plus importante sur une même période de temps pour l'ensemble du réseau.

Outre la consommation globale, l'impact des algorithmes sur les performances du réseau sera principalement évalué sur la mesure de la charge du réseau. Cette mesure se base sur le taux d'utilisation des cliques et permet de représenter l'utilisation des ressources réseau. D'autres mesures viendront compléter ce point par la suite, telles que le taux de paquets délivrés (PDR) et le taux de paquets perdus aux différentes couches réseau lorsque des simulations impliquant le simulateur NS-3 [62] seront réalisées dans le Chapitre 5.

Résultats de simulations

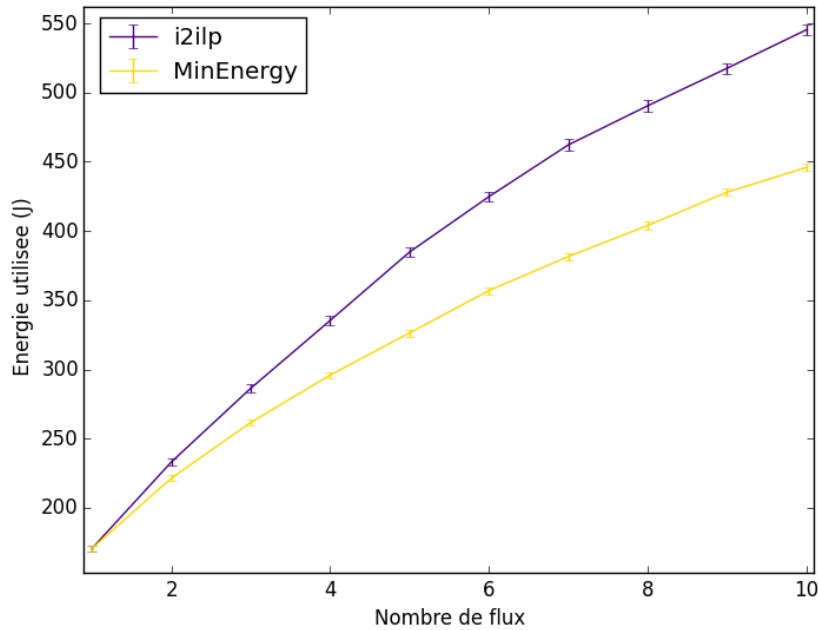


FIGURE 3.4 – Consommation énergétique globale en fonction du nombre de flux.

La Figure 3.4 montre la consommation globale du réseau pendant une unité de temps en fonction du nombre de flux insérés. Comme attendu, notre solution de routage (MinEnergy) est bien plus efficace en termes d'économie d'énergie que l'algorithme de routage sur le plus court chemin, I2ILP (cf. page 30). Naturellement, la consommation globale croît en fonction de la charge du réseau, puisque les nœuds ont davantage de données à transmettre/recevoir (augmentant le temps passé en mode Tx/Rx) et que le nombre de nœuds actifs (sources, destinations, relais) augmente. Or, d'après notre modèle énergétique, un nœud qui est sollicité pour transmettre/recevoir des données n'a pas le droit de se mettre en veille durant la portion de temps où il est inactif.

La Figure 3.5 montre la charge moyenne du réseau en fonction du nombre de flux qui y sont insérés. Nous pouvons voir que notre solution minimisant la consommation énergétique globale charge davantage le réseau que l'algorithme de plus court chemin. Par conséquent, bien que MinEnergy garantisse la QoS demandée en termes de débit (tous les flux peuvent transmettre au débit requis), la capacité du réseau à transmettre des données est réduite à cause de la surcharge provoquée par l'économie d'énergie. Ceci suit toutefois le paradigme selon lequel tout processus d'économie d'énergie a un impact négatif sur les performances d'un système. La difficulté consiste donc à minimiser cet impact pour proposer des solutions intéressantes à appliquer.

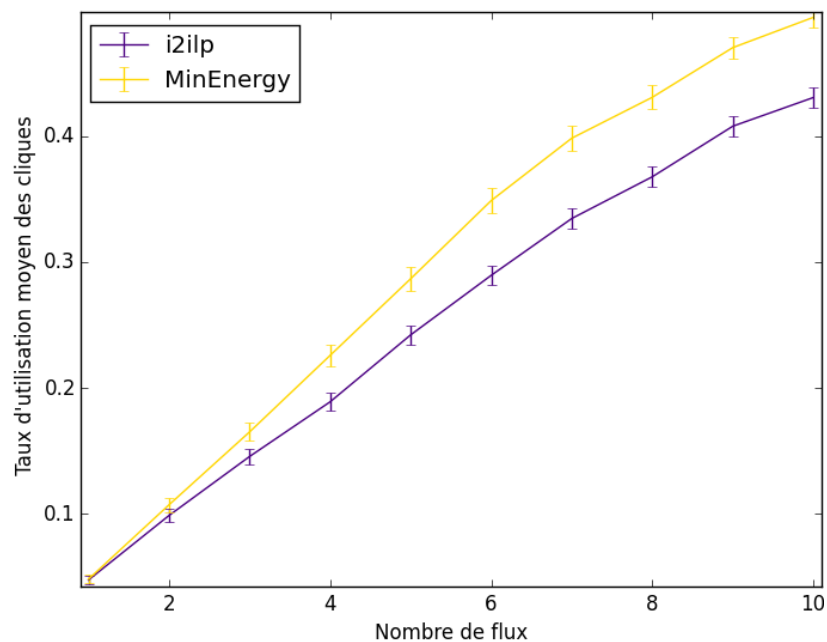


FIGURE 3.5 – Taux d'utilisation moyen des cliques en fonction du nombre de flux

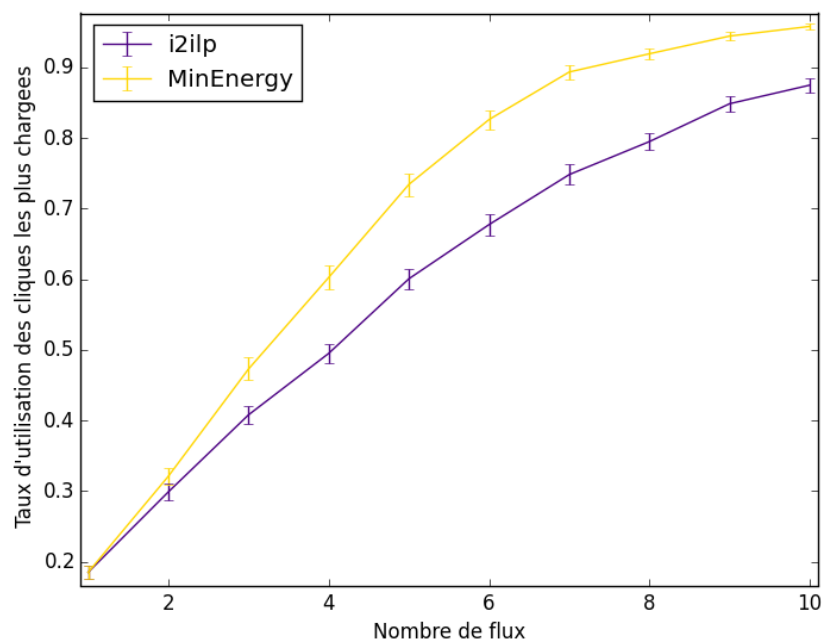


FIGURE 3.6 – Taux d'utilisation de la clique la plus chargée en fonction du nombre de flux.

La Figure 3.6 met justement en avant un point sur lequel nous nous concentrerons par la suite pour, d'une part, réduire l'impact de notre solution de routage sur la capacité du réseau et, d'autre part, proposer une nouvelle approche afin de développer une solution applicable dans des algorithmes existants. Cette figure montre le taux d'utilisation de la (ou des) clique(s) la plus chargée dans le réseau. Nous pouvons voir

3.4 Minimisation du nombre de nœuds actifs

qu'avec notre solution, le taux d'utilisation de la clique la plus chargée est plus élevé qu'avec l'algorithme de routage de plus court chemin. Ceci confirme donc, s'il en était nécessaire, que la minimisation de la consommation énergétique tend à concentrer la charge du réseau sur un nombre limité de nœuds de manière à maximiser le nombre de nœuds pouvant rester en mode veille. Cette confirmation est renforcée par la Figure 3.7 montrant le nombre de nœuds qui ne sont pas utilisés pour router les flux. Il y a donc bien un lien entre le nombre de nœuds actifs dans un réseau et la consommation énergétique globale. C'est cette approche que nous avons choisi de développer et qui est présentée dans la section suivante.

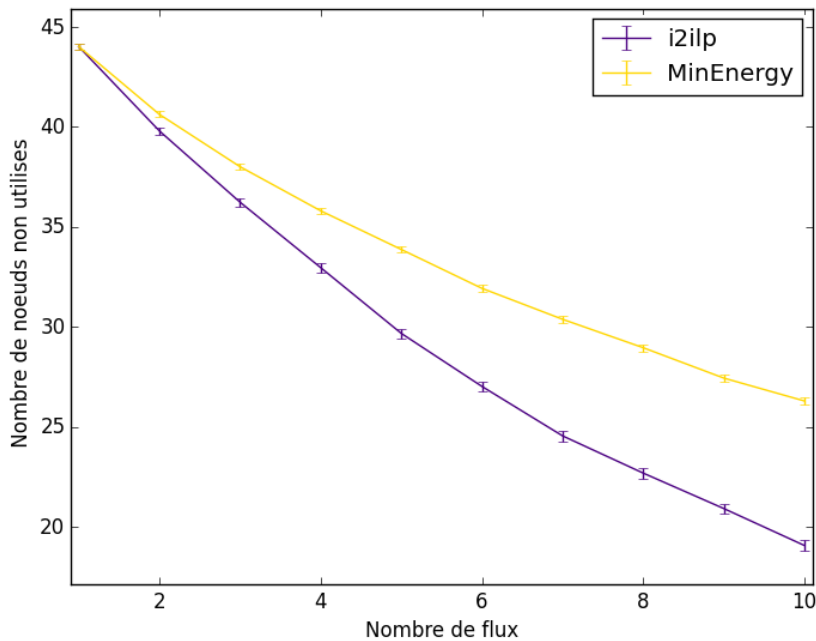


FIGURE 3.7 – Nombre de nœuds utilisés en fonction du nombre de flux.

3.4 Minimisation du nombre de nœuds actifs

Nous avons pu observer que la minimisation de la consommation énergétique globale est liée au nombre de nœuds actifs, c'est-à-dire sollicités pour router des flux dans le réseau. Dans cette partie, nous remplaçons donc la fonction objectif de routage par une fonction de minimisation du nombre de nœuds actifs et montrons l'équivalence.

3.4.1 Programme d'optimisation - Routage avec Agrégation de flux (RA)

Dans la Section 2.4, nous avons établi l'agrégation de flux comme le fait de rassembler les flux d'un réseau sur des chemins communs afin de minimiser le nombre de nœuds actifs dans le réseau. La programmation linéaire permet de modéliser ce comportement, en remplaçant simplement la fonction objectif du programme de routage par une fonction minimisant le nombre de nœuds utilisés. En considérant la variable booléenne x_i , valant 1 si le nœud i est utilisé et 0 sinon, la fonction comptant le nombre de nœuds actifs est la suivante : $\sum_{i \in V} x_i$.

Le programme d'optimisation linéaire routant un ensemble de flux dans un réseau en minimisant le nombre de nœuds utilisés peut alors se modéliser ainsi :

$\text{Min } \sum_{i \in V} x_i$ <p>s.c</p> <ol style="list-style-type: none"> (1) $\sum_{(i,j) \in c} \frac{d_{i,j}}{C_{i,j}^f} \leq 1, \forall c \in C$ (2) $\sum_{(i,j) \in E} x_{i,j}^f - \sum_{(j,k) \in E} x_{j,k}^f = 0, \forall f \in F, \forall j \in V - \{S_f, D_f\}$ (3) $\sum_{(i,S_f) \in E} x_{i,S_f}^f = 0, \forall f \in F$ (4) $\sum_{(S_f,i) \in E} x_{S_f,i}^f = 1, \forall f \in F$ (5) $\sum_{(D_f,i) \in E} x_{D_f,i}^f = 0, \forall f \in F$ (6) $\sum_{(i,D_f) \in E} x_{i,D_f}^f = 1, \forall f \in F$ (7) $\sum_{f \in F} x_{i,j}^f \cdot d_f = d_{i,j}, \forall (i,j) \in E$ (8) $\sum_{(i,j) \in E} x_{i,j}^f \leq 1, \forall f \in F, \forall i \in V - \{D_f\}$ (9.1) $\text{Max}_{(i,j) \in E} (C_{i,j}) \cdot x_i - \sum_{(i,j) \in E} d_{i,j} \geq 0, \forall i \in V$ (9.2) $\text{Max}_{(j,i) \in E} (C_{j,i}) \cdot x_i - \sum_{(j,i) \in E} d_{j,i} \geq 0, \forall i \in V$ (9.3) $\sum_{j \in V} (d_{i,j} + d_{j,i}) - \text{Min}_{f \in F} (d_f) \cdot x_i \geq 0, \forall i \in V$ (10) $u_i^f - u_j^f + V \cdot x_{i,j}^f \leq V - 1, \forall f \in F, \forall (i,j) \in E [i \neq S_f, j \neq D_f]$ <p> $x_{i,j}^f \in \{0, 1\}, \forall (i,j) \in E, \forall f \in F$ $d_{i,j}^f \in \mathbb{R}^+, \forall (i,j) \in E, \forall f \in F$ $x_i \in \{0, 1\}, \forall i \in V, \forall i \in V$ $u_i^f \in \mathbb{R}^+, \forall f \in F, \forall i \in V - \{S_f\}$ </p>

3.4 Minimisation du nombre de nœuds actifs

Cette formulation a pour variables de décision $d_{i,j}$ (capacité utilisée d'un lien), $x_{i,j}^f$ (la variable de routage indiquant si un flux passe sur un lien), x_i (l'utilisation ou non d'un nœud), et u_i^f qui sera présentée avec la contrainte qui l'introduit. De nouveau, ces variables de décision sont dépendantes les unes des autres et peuvent toutes être dérivées à partir de la valeur qui est attribuée à $x^{i,j}$. A l'instar du programme de minimisation de la consommation énergétique, celui-ci est composé d'une fonction objectif (la minimisation du nombre de nœuds utilisés), des contraintes de routage et des contraintes (9.1), (9.2) et (9.3) d'activation des nœuds. La fonction objectif ne prenant plus en compte la consommation énergétique des nœuds, les contraintes qui y sont liées ne sont plus nécessaires et peuvent donc être supprimées.

En revanche, un problème de boucles, qui était jusqu'à présent résolu grâce aux fonctions objectifs de minimisation des chemins ou de l'énergie, doit désormais être traité explicitement. Il s'agit de boucles sur lesquelles la résolution du programme peut décider de router un flux si cela n'affecte pas la valeur de la fonction objectif de la solution optimale et ne viole pas la contrainte de capacité. Ces boucles ont la particularité d'être déconnectées du chemin qui a été attribué au flux. Pour résoudre ce problème, nous introduisons la contrainte (10), issue de la modélisation du problème du voyageur de commerce [63], afin d'interdire la présence de boucles (quelles soient connectées ou non) dans les chemins attribués aux flux.

3.4.2 Résultats de simulation

Dans les résultats ci-dessous, nous comparons notre solution de minimisation du nombre de nœuds utilisés (RA), qui représente l'agrégation optimale, avec la solution de minimisation de la consommation énergétique globale (MinEnergy) présentée dans la Section 3.3. L'algorithme de plus court chemin I2ILP est également présent pour servir de courbe de référence.

La Figure 3.8 montre la consommation énergétique globale du réseau en fonction du nombre de flux insérés. Nous pouvons remarquer que la courbe associée à notre solution d'agrégation se confond parfaitement avec celle de la minimisation de la consommation énergétique. Notre intuition de départ est donc confirmée : minimiser la consommation énergétique revient à minimiser le nombre de nœuds actifs dans le réseau (ou maximiser le nombre de nœuds éteints).

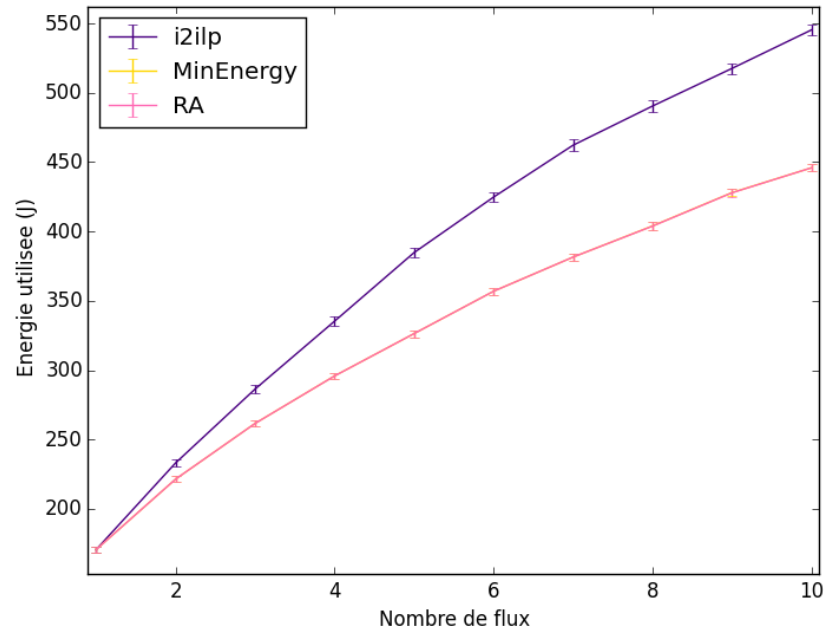


FIGURE 3.8 – Consommation énergétique globale en fonction du nombre de flux insérés.

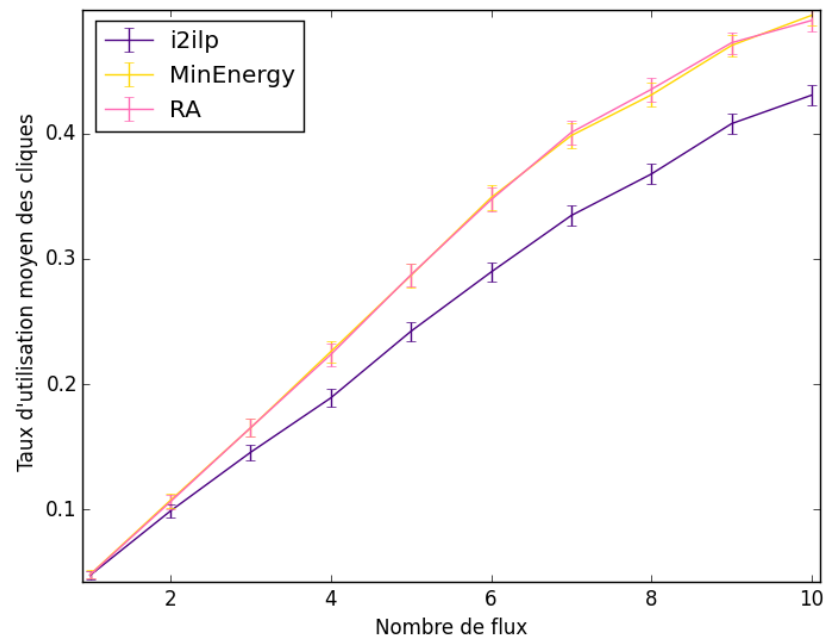


FIGURE 3.9 – Taux d'utilisation moyen des cliques en fonction du nombre de flux .

Les Figures 3.9 et 3.10 viennent confirmer la similarité entre ces deux algorithmes en termes de charge globale (taux d'utilisation moyen des cliques) et de charge locale (taux d'utilisation maximal des cliques). Cela montre aussi que le routage minimisant le nombre de nœuds utilisés dans le réseau souffre des mêmes limites vis-à-vis de la charge que le routage minimisant la consommation énergétique.

3.4 Minimisation du nombre de nœuds actifs

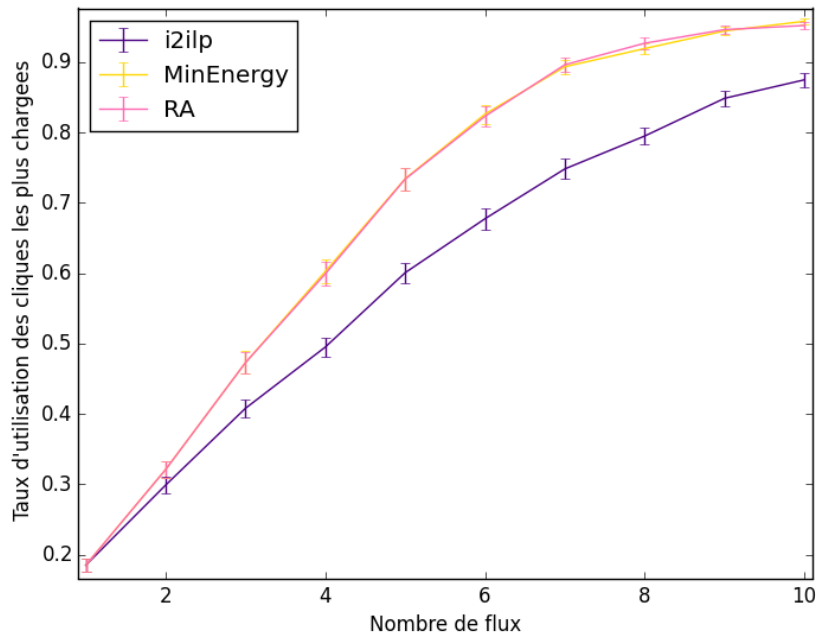


FIGURE 3.10 – Taux d’utilisation de la clique la plus chargée en fonction de nombre de flux.

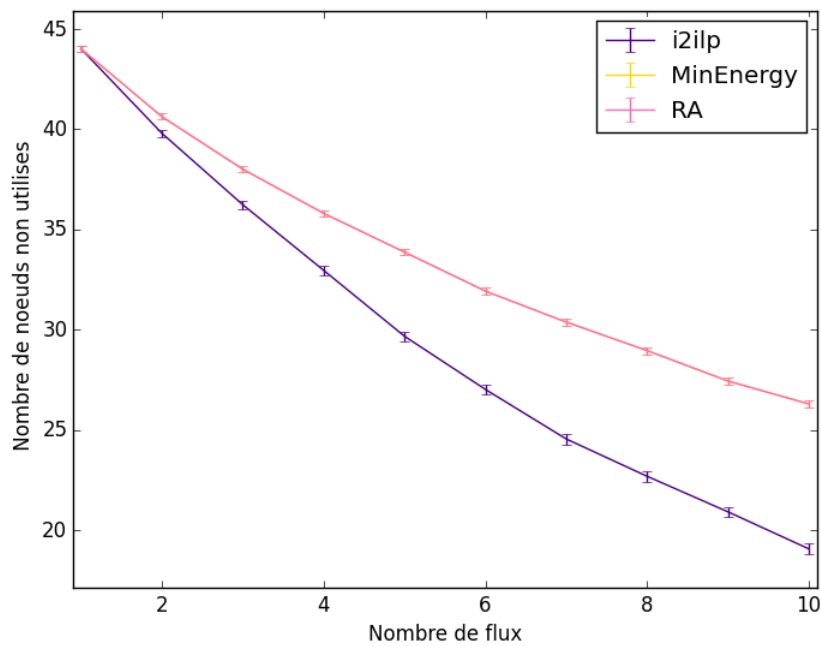


FIGURE 3.11 – Nombre de nœuds utilisés en fonction du nombre de flux.

L'équivalence des solutions se retrouve enfin dans la Figure 3.11 où le nombre de nœuds non utilisés, et donc pouvant être éteints, est identique à la solution de minimisation de la consommation énergétique globale. Ceci peut être expliqué par le fait que, selon nos paramètres de simulation, un nœud allumé (dans l'état Tx, Rx ou Idle) consomme au-delà de huit fois plus d'énergie qu'en veille (état Sleep). Il faut

donc avoir une bonne raison pour allumer un nœud, comme par exemple connecter une source à une destination ou offrir plus de capacité au réseau. Car il est finalement plus profitable à l'ensemble du réseau en termes de consommation énergétique de demander un peu plus de travail de la part des nœuds déjà allumés plutôt que d'en réveiller d'autres.

3.5 Codage Réseau

Si, comme nous l'avons vu dans la Section 3.4, l'agrégation de flux semble être une solution intéressante pour réduire la consommation énergétique globale des réseaux sans fil multi-sauts, nous avons également observé que concentrer le trafic sur un trop petit nombre de nœuds peut amener à une surcharge des cliques du réseau. Cette partie montre comment exploiter cette agrégation de flux afin de réduire le nombre de transmissions nécessaires au transfert des paquets des flux en faisant du codage réseau. La synergie offerte par l'utilisation conjointe de l'agrégation de flux et du codage réseaux permet ainsi d'améliorer l'agrégation et de réduire la charge des cliques.

3.5.1 Introduction au codage réseau

Le codage réseau est un paradigme de routage permettant aux nœuds intermédiaires de manipuler les paquets avant de les retransmettre le long de leur chemin afin de réduire le nombre de transmissions et/ou de retransmissions selon la méthode de codage utilisée. Par exemple, le codage réseau inter-flux permet à un nœud intermédiaire de combiner les paquets de plusieurs flux afin de réduire le nombre de transmissions nécessaires au transfert des données de ces flux.

Ce principe peut être illustré par un simple modèle de communication du type Alice-Bob (c.f. Figure 3.12). Soit Alice (A) et Bob (B) souhaitant chacun envoyer un message à l'autre, par le biais d'un nœud intermédiaire (Relai). Dans un premier temps, A et B transmettent leur message au relai. Sans codage réseau, le relai retransmet alors les messages à leur destination respective, l'un après l'autre, sans effectuer d'opération particulière. Les données ont ainsi pu être échangées après quatre transmissions. Avec du codage réseau, le relai peut combiner les messages entre eux, en réalisant par exemple une simple opération XOR et diffuser en une seule transmission cette combinaison à A et B. Ces derniers peuvent alors décoder le message qui leur est destiné en réalisant la même opération sur la combinaison reçue et le message qu'ils ont eux-mêmes envoyé. Les données ont donc pu être échangées avec trois transmissions seulement, réalisant

3.5 Codage Réseau

ainsi un gain de 25%.

Cette méthode est utilisée dans [45] pour proposer une solution, basée sur un programme d'optimisation linéaire, permettant de router un ensemble de flux en minimisant le nombre de transmissions dans le réseau. Toutefois, cette solution ne prend pas en compte les interférences liées aux communications sans fil. Dans [46], une contrainte de cliques est ajoutée au modèle afin d'améliorer la capacité et le taux d'acceptation des flux.

L'efficacité du codage réseau dépend naturellement des opportunités de codage dans le réseau. Or notre approche permet justement de créer de telles opportunités. En agrégeant les flux sur un ensemble réduit de nœuds, nous augmentons la probabilité d'apparition des conditions nécessaires à l'application du modèle de codage réseau illustré par la Figure 3.12.

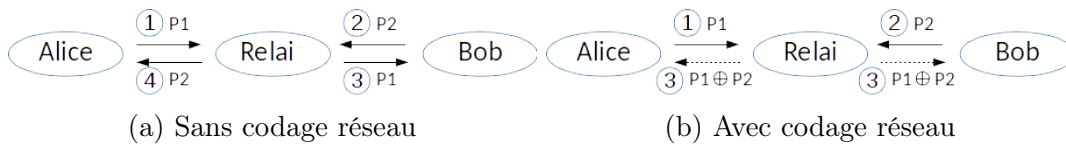


FIGURE 3.12 – Modèle de codage réseau de type Alice-Bob.

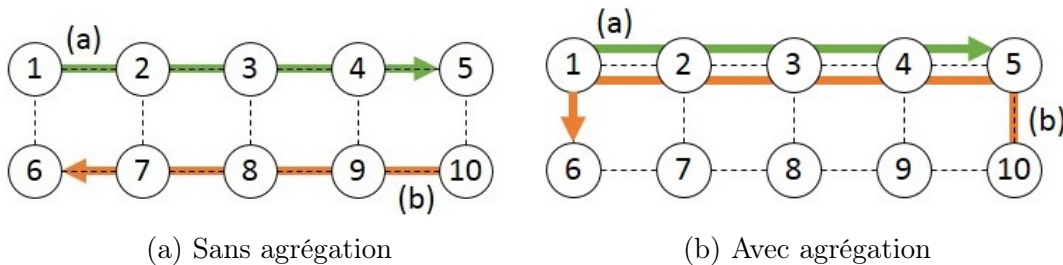


FIGURE 3.13 – Exemple de deux flux suivant un plus court chemin ou étant agrégés.

La Figure 3.13 montre l'effet de l'agrégation de flux sur le routage et illustre l'intérêt d'exploiter cette situation pour appliquer du codage réseau. Supposons que nous ayons deux flux a et b , chacun ayant une demande de 10, à router sur un réseau en grille composé de dix nœuds, avec des liens de 100 de capacité. Une première solution de routage consiste à router les flux sur le plus court chemin. Ainsi, le flux a est routé sur le chemin $\{1, 2, 3, 4, 5\}$ et le flux b sur le chemin $\{10, 9, 8, 7, 6\}$. Pour vérifier la charge du réseau, il convient de définir le graphe de conflit associé dans cette situation. Nous le représentons sur la Figure 3.14.

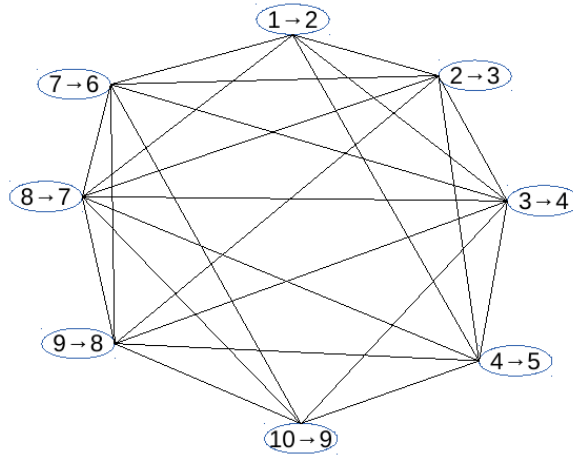


FIGURE 3.14 – Graphe de conflit de la topologie sans agrégation.

À partir de ce graphe de conflit, que nous avons construit en considérant uniquement les liens utilisés afin de l'alléger (les liens non utilisés ne générant pas d'interférence), nous déterminons les cliques (c'est-à-dire les ensembles maximaux de liens interférant) :

- $C_1 = \{(1, 2), (2, 3), (3, 4), (4, 5), (8, 7)\}$
- $C_2 = \{(7, 6), (8, 7), (9, 8), (10, 9), (3, 4)\}$
- $C_3 = \{(3, 4), (4, 5), (8, 7), (9, 8), (10, 9)\}$
- $C_4 = \{(2, 3), (3, 4), (4, 5), (8, 7), (9, 8)\}$

Il nous est alors possible de calculer le taux d'utilisation de chacune de ces cliques en utilisant la formule présentée dans la partie 3.1.2. Ainsi, pour C_1 , il est égal à :

$$\begin{aligned} & \frac{d_{1,2}}{C_{1,2}} + \frac{d_{2,3}}{C_{2,3}} + \frac{d_{3,4}}{C_{3,4}} + \frac{d_{4,5}}{C_{4,5}} + \frac{d_{8,7}}{C_{8,7}} \\ &= \frac{10}{100} + \frac{10}{100} + \frac{10}{100} + \frac{10}{100} + \frac{10}{100} \\ &= 50\% \end{aligned}$$

Nous trouvons la même valeur pour les autres cliques. Tous les flux pourraient donc augmenter leur demande jusqu'à 20 unités, sans surcharger le réseau.

En supposant maintenant que les deux flux soient agrégés, b est désormais routé sur le chemin $\{10, 5, 4, 3, 2, 1, 6\}$. Les liens utilisés pour router les deux flux ayant changé, le graphe de conflit correspondant est représenté par la Figure 3.15.

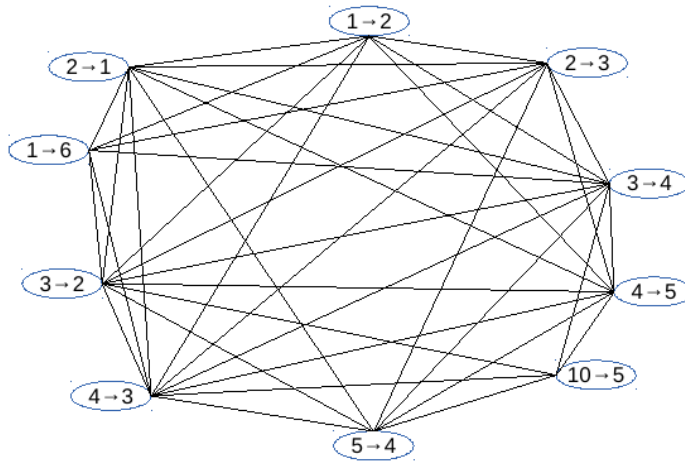


FIGURE 3.15 – Graphe de conflit de la topologie avec agrégation.

On peut ensuite en déduire les cliques impliquées dans ce nouveau routage :

- $C_1 = \{(1, 2), (2, 3), (3, 4), (4, 5), (3, 2), (4, 3), (2, 1)\}$
- $C_2 = \{(2, 1), (3, 2), (4, 3), (5, 4), (2, 3), (3, 4), (4, 5)\}$
- $C_3 = \{(1, 6), (2, 1), (3, 2), (4, 3), (1, 2), (2, 3), (3, 4)\}$
- $C_4 = \{(10, 5), (3, 4), (4, 5), (3, 2), (4, 3), (5, 4)\}$

Le calcul de la charge des cliques montre un taux d'utilisation de 70% pour les cliques C_1 , C_2 , C_3 et de 60% pour la clique C_4 . Nous sommes donc encore en mesure de satisfaire les demandes originelles des flux, mais ces dernières ne peuvent désormais monter que jusqu'à 14,28. La capacité du réseau a donc diminué, conformément aux résultats présentés dans la Section 3.4. Ceci s'explique par le fait qu'en agrégeant les flux, certaines cliques se retrouvent davantage chargées à cause du nombre de transmissions qui augmente. En effet, les transmissions sur les liens (10, 5) et (1, 6), qui ont aidé au reroutage du flux b , imposent une charge supplémentaire dans le réseau par rapport au routage sur le plus court chemin. Par conséquent, nous avons la possibilité d'éteindre trois nœuds pour une économie d'environ 30%, mais cela a un coût non négligeable sur la capacité globale du réseau.

C'est donc pour réduire ce coût en termes de capacité que le codage réseau peut être utilisé. En effet, au niveau des nœuds 2, 3 et 4, le codage inter-flux présenté précédemment peut être appliqué pour réduire le nombre de transmissions dans le réseau. Lorsque des paquets sont en attente d'être transmis pour chacun des nœuds,

un total de trois transmissions peut être économisé sur l'ensemble du réseau. Ce gain d'une transmission par rapport au routage classique (plus court chemin) nous permet donc d'espérer, dans certains cas, un gain en termes de consommation énergétique, mais aussi de capacité globale, lorsque l'agrégation de flux est associée au codage réseau. Si nous reprenons l'exemple précédent, le calcul de la charge des cliques (en considérant que le codage réseau permet de réduire le nombre de transmissions, cf. partie 3.5.2) montre que les flux peuvent augmenter leur demande jusqu'à un débit de 25, soit un gain de performance de 25% par rapport au plus court chemin.

3.5.2 Application à l'agrégation de flux (RANC et RA+NC)

Nous avons deux manières d'intégrer le codage réseau à l'agrégation de flux. La première consiste à modéliser le codage réseau par un bloc de contraintes à ajouter dans le programme de routage avec agrégation (RANC). Ainsi, en prenant en compte l'effet du codage réseau sur la capacité du réseau, il devient possible de réduire encore davantage le nombre de nœuds utilisés et donc d'améliorer l'agrégation de flux. La contrainte de cliques avec prise en compte du codage réseau, présentée dans [46], est adaptée à notre modèle comme suit :

$$\sum_{(i,j) \in c} \frac{d_{i,j}}{C_{i,j}} - \sum_{(i,j),(i,k) \in c, k < j} c_{k,i,j} \cdot \left(\frac{1}{2C_{i,j}} + \frac{1}{2C_{i,k}} \right) \leq 1, \forall c \in C$$

Cette contrainte assure que le taux d'utilisation des cliques ne dépasse pas 100%, en considérant la réduction de charge induite sur les cliques grâce au codage réseau. Ici, la variable $c_{k,i,j}$ représente la quantité de trafic qui peut être codé au niveau du nœud i venant du lien (k, i) et ayant pour prochain saut le nœud j . Cette quantité de trafic pouvant être codé $c_{k,i,j}$ est définie par :

$$\begin{aligned} c_{k,i,j} - w_{k,i,j} &\leq 0, \forall (i, j), (k, i) \in E \\ c_{k,i,j} - w_{j,i,k} &\leq 0, \forall (i, j), (k, i) \in E \\ c_{k,i,j} - c_{j,i,k} &= 0, \forall (i, j), (k, i) \in E \end{aligned}$$

Ces contraintes utilisent la quantité totale de trafic allant du nœud k au nœud j (et inversement) et passant par le nœud intermédiaire i ($w_{k,i,j}$ et $w_{j,i,k}$) pour en déduire la quantité maximale de trafic que le nœud i est autorisé à coder sur ce chemin $\{k, i, j\}/\{j, i, k\}$. Cette valeur est égale à $\min(w_{j,i,k}, w_{k,i,j})$, étant donné qu'il n'est pas possible de coder plus de trafic qu'il n'en arrive en i . De plus, contrairement à [46], la maximisation du codage sur ce chemin ne peut pas être obtenue par la maximisation de la fonction objectif. Cette propriété est garantie par les contraintes suivantes, qui assurent que la quantité de trafic codé par le nœud i pour le chemin $\{k, i, j\}$ est

3.5 Codage Réseau

supérieure ou égale à $\min(w_{j,i,k}, w_{k,i,j})$. La combinaison avec les contraintes précédentes permet d'avoir l'égalité $c_{k,i,j} = \min(w_{j,i,k}, w_{k,i,j})$.

$$\begin{aligned} c_{k,i,j} - w_{k,i,j} + C_{i,j} \cdot t_{k,i,j} &\geq 0, \forall (i,j), (k,i) \in E \\ c_{k,i,j} - w_{j,i,k} + C_{i,j} \cdot (1 - t_{k,i,j}) &\geq 0, \forall (i,j), (k,i) \in E \end{aligned}$$

Enfin, les contraintes suivantes permettent de lier les contraintes de codage au reste du programme d'optimisation du routage avec agrégation, en calculant la quantité de trafic $w_{i,j,k}$ qui circule sur une portion de chemin $\{i, j, k\}$.

$$\begin{aligned} (\sum_{f \in F} m_{i,j,k}^f) - w_{i,j,k} &= 0, \forall (i,j), (j,k) \in E \\ (\sum_{(j,k) \in E} m_{i,j,k}^f) - x_{i,j}^f \cdot M_f &= 0, \forall (i,j) \in E, \forall f \in F \\ (\sum_{(i,j) \in E} m_{i,j,k}^f) - x_{j,k}^f \cdot M_f &= 0, \forall (j,k) \in E, \forall f \in F \end{aligned}$$

La seconde utilisation du codage réseau consiste à l'appliquer après que l'algorithme d'agrégation de flux a déterminé les routes. RA+NC est donc un algorithme qui se déroule en deux temps. La première étape consiste à router les flux en résolvant le programme d'optimisation RA. Ensuite, le calcul des mesures (taux de charge des cliques et consommation énergétique) est réalisé en prenant en compte le codage réseau. De cette manière, le codage réseau est uniquement utilisé pour réduire l'impact de l'agrégation de flux sur la charge globale du réseau et la consommation énergétique.

3.5.3 Résultats des simulations

Nous évaluons les performances des deux algorithmes de routage avec agrégation de flux et codage réseau, RANC (le codage réseau est intégré dans le programme d'optimisation) et RA+NC (le codage réseau est appliqué une fois le routage établi par le programme de routage avec agrégation optimale). Les résultats sont comparés à ceux obtenus avec le programme d'agrégation optimale de flux RA, décrit dans la partie 3.4. Nous présentons aussi l'algorithme de plus court chemin I2ILP en tant que mesure de référence, pour évaluer l'effet du codage réseau sur la réduction de la capacité globale causée par l'agrégation de flux.

La Figure 3.16 montre le nombre de nœuds pouvant être éteints en fonction du nombre de flux insérés dans le réseau. Nous pouvons remarquer une légère amélioration lorsque le codage réseau est pris en compte lors du calcul des routes (RANC) par rapport à la solution optimale (RA). Ceci est dû au fait que le codage réseau, en réduisant le nombre de transmissions, et par extension la charge du réseau, offre l'opportunité d'agréger des flux qui ne pouvaient pas l'être avec RA à cause de la surcharge

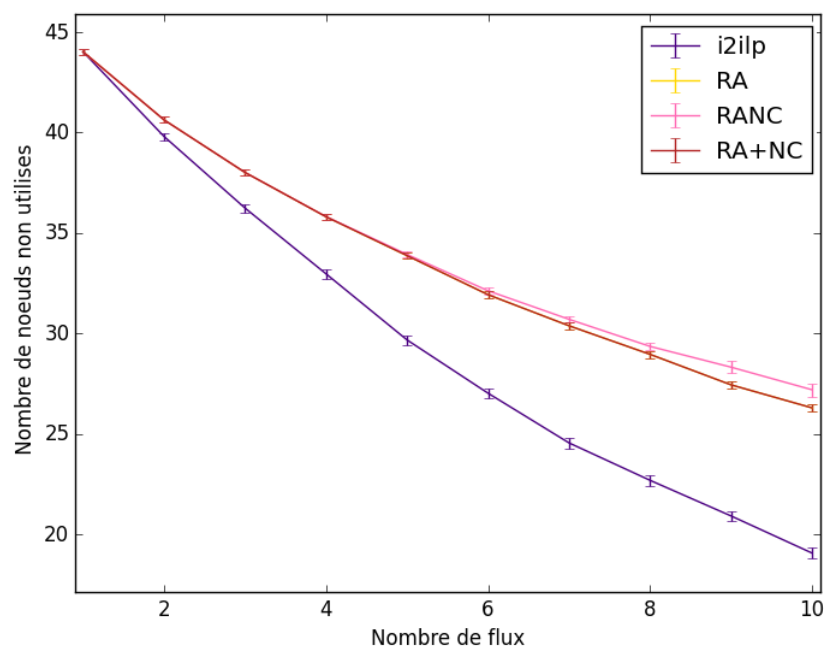


FIGURE 3.16 – Nombre de nœuds utilisés en fonction du nombre de flux.

des cliques. Bien entendu, comme RA+NC est un algorithme se basant sur le résultat de RA et que le codage réseau n'a, dans ce cas, pas d'influence sur l'agrégation des flux, les deux courbes se superposent. Toutefois, nous remarquons que l'amélioration en termes d'agrégation de flux est extrêmement faible avec RANC par rapport à la complexité en nombre de variables et de contraintes (de l'ordre de $O(|F| \cdot |V| \cdot \text{DegreMax}^2)$) qu'implique la prise en compte du codage dans le programme d'optimisation.

Les Figures 3.17 et 3.18 montrent l'effet du codage réseau sur la charge globale et locale du réseau. Dans les deux cas, le codage permet de réduire l'impact de l'agrégation des flux sur la charge moyenne et maximale des cliques. Lorsque le codage réseau est utilisé uniquement pour diminuer la charge du réseau (RA+NC), la charge supplémentaire générée par l'agrégation de flux peut être réduite de plus de la moitié, ce qui n'est pas le cas lorsque le codage réseau est utilisé pour améliorer l'agrégation. Par ailleurs, au fur et à mesure que nous ajoutons des flux dans le réseau, l'algorithme RA+NC tend à se rapprocher des performances du plus court chemin, alors que RANC tend vers le niveau de l'agrégation optimale de flux, annulant ainsi les bénéfices du codage réseau.

Par conséquent, un choix doit être fait quant à la méthode à retenir pour l'implémentation d'une méthode de routage applicable. En comparant la solution de routage optimisant l'agrégation des flux grâce au codage réseau et celle, sous-optimale, ayant un impact très limité sur la baisse de performance du réseau, la seconde est à privilégier.

3.5 Codage Réseau

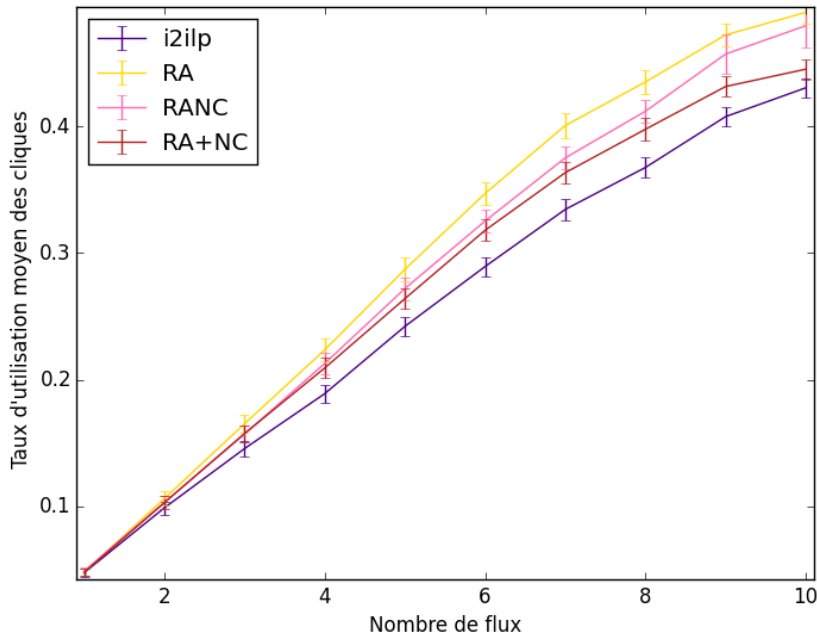


FIGURE 3.17 – Taux d’utilisation moyen des cliques en fonction du nombre de flux insérés.

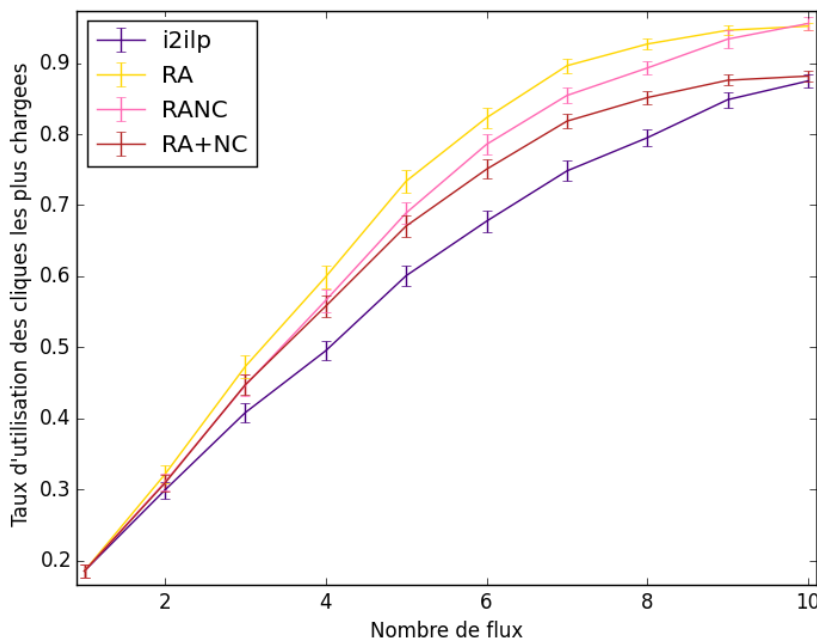


FIGURE 3.18 – Taux d’utilisation de la clique la plus chargée en fonction du nombre de flux.

La raison principale venant justifier ce choix est la séparation de ces deux paradigmes. L’agrégation de flux et le codage réseau pouvant profiter l’un de l’autre et étant des processus complexes à mettre en œuvre, il est plus simple et plus intéressant de les développer séparément pour ensuite profiter de leurs avantages couplés. Une autre raison relève de l’objectif initial de l’utilisation du codage réseau. Cette technique est

naturellement apparue appropriée pour réduire l'impact de l'agrégation de flux sur la charge du réseau. Elle n'a pas vocation à influencer l'algorithme de routage. D'ailleurs, l'amélioration en termes d'agrégation apportée par RANC est bien trop faible par rapport à l'explosion de la complexité du programme linéaire qui en résulte et du temps nécessaire à sa résolution. Ce constat nous a donc permis de déterminer la direction prise durant cette thèse et de nous focaliser sur le problème d'agrégation de flux, tout en sachant que la baisse éventuelle de capacité du réseau peut être compensée (voire inversée) grâce à l'utilisation du codage réseau.

3.6 Conclusion

La réduction de la consommation énergétique globale des réseaux sans fil multi-sauts peut être réalisée par le routage et se rapporter à un problème de routage minimisant le nombre de nœuds utilisés. Nous avons présenté une solution de routage se basant sur un critère indépendant des paramètres énergétiques et optimale en termes de consommation globale du réseau. Il est donc possible d'améliorer l'efficacité énergétique des réseaux sans fil multi-sauts sans passer par des solutions multi-couches utilisant des mesures de consommation énergétique ou d'énergie résiduelle. Cette approche est cependant source d'une augmentation de la charge du réseau que nous avons caractérisée par une hausse du taux d'utilisation des cliques. Toutefois, nous avons vu que cette agrégation de flux offre des opportunités permettant de réduire le nombre de transmissions grâce au codage réseau. Ceci permet donc de limiter (voire d'inverser) l'impact de l'agrégation sur la capacité du réseau à transmettre des flux de données.

En revanche, les méthodes présentées dans ce chapitre ne sont pas applicables en l'état. En effet, la résolution des programmes linéaires en nombres entiers étant NP-difficile, ce genre de solution n'est pas envisageable pour une application en temps réel dans des protocoles de routage existants. De plus, nos solutions se rapportent à du routage centralisé et ne sont donc pas, par nature, adaptées aux réseaux sans fil multi-sauts. Le chapitre suivant présente donc une nouvelle contribution pour résoudre ce problème et ainsi permettre à des algorithmes de routage existants de réaliser de l'agrégation de flux.

3.6 Conclusion

Chapitre 4

Métrieque d'agrégation de flux : FAME

La solution optimale présentée dans le Chapitre 3 repose sur la programmation linéaire en nombres entiers et n'est donc, en l'état, pas applicable dans un réseau réel. De plus, la méthode de modélisation et la résolution du problème requiert une solution centralisée, ce qui n'est pas adaptée aux réseaux sans fil multi-sauts, par nature. Nous avons vu dans le Chapitre 2 que les protocoles de routage dans les réseaux sans fil multi-sauts reposent, la plupart du temps, sur une implémentation distribuée d'un algorithme de routage avec minimisation du coût de routage. Le plus court chemin n'est en réalité qu'un cas particulier de ce type de routage dont le comportement peut être modifié selon la métrieque utilisée.

Nous avons présenté dans la Section 2.1 que des métrieques de routage telles que MTPR, MMBCR et MDR [19, 20, 21] permettaient d'améliorer l'efficacité énergétique des réseaux sans fil multi-sauts en termes de durée de vie. Toutefois, ces métrieques tendent à répartir la charge du réseau sur l'ensemble des nœuds de manière équitable, afin d'éviter l'utilisation excessive de certains nœuds centraux. Hors il a été montré dans le chapitre précédent que cette stratégie n'est pas efficace en termes de consommation énergétique globale. De plus, ces métrieques sont principalement basées sur des données issues d'autres couches que le niveau réseau. Or le concept d'étanchéité des couches protocolaires est une notion que nous souhaitons respecter lors de la conception des protocoles réseaux, afin de garantir un maximum d'interopérabilité.

Ce chapitre présente donc la métrieque d'agrégation de flux, que nous appellerons FAME (Flow Aggregation MEtric). Cette métrieque peut être intégrée de manière transparente dans n'importe quel algorithme de routage de plus court chemin (pour peu qu'elle ait été préalablement calculée) et permet de remplacer l'objectif de minimisation de la longueur des chemins par la celle du nombre de nœuds utilisés dans le réseau.

4.1 Métrique d'agrégation de Flux : FAME

Dans un premier temps, nous définissons cette métrique, la manière de la calculer et son intégration dans un algorithme de routage de plus court chemin. Dans un second temps, nous présentons une solution permettant d'approximer la contrainte de clique afin de préserver la QoS et ainsi éviter de surcharger le réseau au-delà de sa capacité.

4.1 Métrique d'agrégation de Flux : FAME

FAME est une métrique de routage permettant de calculer le coût d'utilisation des liens d'un réseau sans fil multi-sauts, de sorte qu'un algorithme de routage de flux sur le plus court chemin puisse agréger ces flux sur un ensemble défini de nœuds. Ce dernier peut être déterminé à l'avance, lors du déploiement du réseau, ou dynamiquement en fonction de l'état courant du réseau. Nous présentons donc le processus qui nous a permis d'aboutir à cette approche. Puis nous définissons cet ensemble de nœuds vers lesquels les flux doivent être agrégés et discutons des différentes manières de le construire. Enfin, nous détaillons le calcul de la métrique avant de conclure en présentant les performances de notre solution en termes d'agrégation et de charge des cliques.

4.1.1 Intuition

Afin d'élaborer une stratégie de routage applicable pour agréger des flux, il est nécessaire d'identifier les critères permettant de sélectionner les nœuds intermédiaires chargés de relayer les flux. La Figure 4.1 illustre trois exemples de flux aléatoires routés sur trois topologies différentes. Ils nous permettent de comparer les chemins empruntés lorsque les flux sont routés avec le plus court chemin (à gauche) et avec la solution d'agrégation optimale (à droite), présentée dans le Chapitre 3.13.

Tout d'abord, nous pouvons remarquer que que les flux agrégés semblent suivre des chemins formant un arbre couvrant, voire même un arbre de Steiner [64], entre les sources et les destinations. Mais baser le routage des flux sur un arbre de Steiner nécessite le développement d'un algorithme spécifique, ce qui est contraire à notre objectif de proposer une solution générique et intégrable dans les algorithmes de routage existants. De plus, nous voulons développer une solution de routage basée sur un algorithme de plus court chemin classique et autorisant l'utilisation d'une métrique permettant l'agrégation.

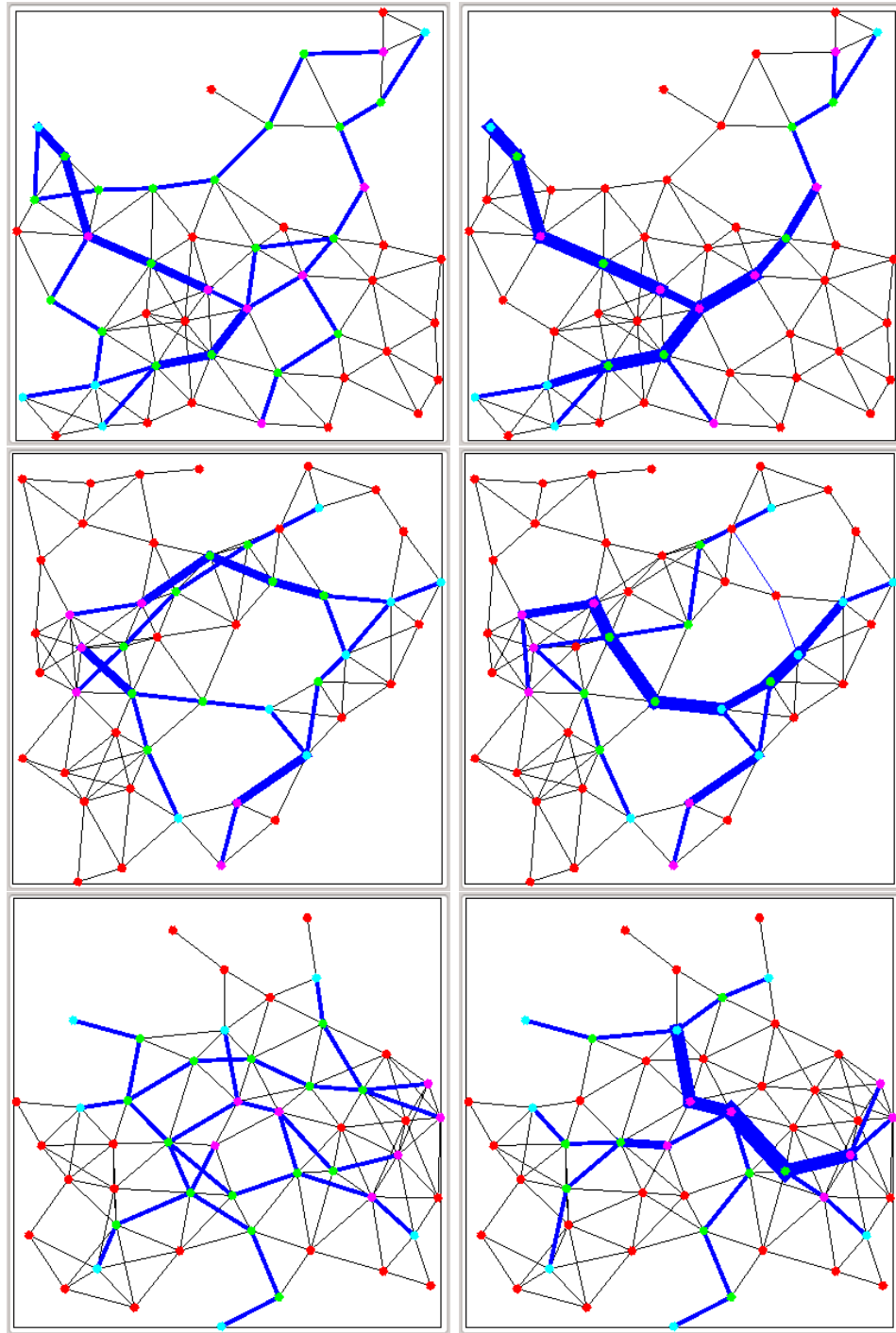


FIGURE 4.1 – Comparaison entre le plus court chemin (à gauche) et l'agrégation optimale (à droite).

Les exemples montrent aussi que la part de nœuds sources et destinations (cyan et magenta) parmi les nœuds utilisés est bien plus grande lorsque nous agrégeons les flux. Par conséquent, les nœuds sources et destinations ont plus de chance d'être utilisés comme nœuds intermédiaires pour router les autres flux. Il existe donc des nœuds dans le réseau par lesquels il est préférable de concentrer les flux.

4.1 Métrique d'agrégation de Flux : FAME

L'objectif, à terme, est de proposer une solution applicable pouvant s'intégrer dans les algorithmes existants afin de limiter au maximum son coût de déploiement. Agréger les flux sur les sources et les destinations semble être le choix le plus judicieux. Ceci peut se justifier par le fait que, quelle que soit la méthode de routage sélectionnée, ces nœuds devront être utilisés pour envoyer ou recevoir les flux. Il est donc intéressant de les réutiliser au maximum en tant que nœuds intermédiaires pour router les autres flux du réseau. Mais ce n'est pas le seul choix possible et d'autres nœuds peuvent éventuellement être utilisés en fonction des besoins ou des caractéristiques du réseau sur lequel notre solution est appliquée.

4.1.2 Nœuds d'intérêt

Comme nous l'avons observé dans la partie précédente, nous souhaitons router un ensemble de flux de manière à donner le rôle de relais en priorité à un ensemble de nœuds définis. Nous appelons un nœud composant cet ensemble un Nœud d'Intérêt (*NI*). Puisque notre approche se base sur l'utilisation d'une métrique représentant le coût de routage relatif à l'utilisation des nœuds intermédiaires, diriger les flux vers ces derniers revient à leur attribuer un coût (poids) suffisamment faible. Ainsi, l'algorithme de plus court chemin considère qu'il est plus intéressant de router les flux sur eux, ce qui mène l'agrégation des flux sur ces nœuds d'intérêt et leur voisinage.

Dans cette thèse, et pour les raisons citées ci-dessus, nous considérons les sources et destinations comme étant les nœuds d'intérêt, à partir desquels nous calculons le poids des nœuds du graphe. Ce choix nous permet de calculer une métrique de routage dynamique pouvant s'adapter à la mobilité des flux. Mais d'autres critères peuvent également être utilisés pour construire l'ensemble des *NI*s. Considérer un ensemble dominant, connecté (CDS : *connected dominating set*) ou non, ou un arbre de Steiner, permettrait d'éviter de diffuser l'information sur les sources et les destinations du réseau. Mais cela aurait pour conséquence de créer un ensemble de *NI*s trop grand et donc de réduire significativement l'efficacité de l'agrégation. Les nœuds d'intérêt peuvent aussi être sélectionnés de façon statique lors du déploiement du réseau. Cette approche serait pertinente dans le cas de réseaux hétérogènes composés de nœuds plus efficaces énergétiquement ou disposant de plus de ressources (réseau ou calcul) sur lesquels il peut donc être intéressant de concentrer les flux, ou dans des réseaux aux propriétés particulières (points d'accès pour des réseaux mesh, puits dans les réseaux de capteurs, etc.).

4.1.3 Calcul de la métrique

Notre calcul de la métrique FAME considère deux cas. Les nœuds d'intérêt sont les nœuds sur lesquels nous voulons router les flux en priorité. Il faut donc leur attribuer un poids faible afin d'augmenter les chances que ces nœuds soient sélectionnés en tant que nœuds intermédiaires. Nous fixons cette valeur à $\frac{1}{|V|-1}$, ce qui représente, comme nous allons le voir, le plus petit poids qu'un nœud n'appartenant pas à l'ensemble des *NI*s puisse avoir théoriquement. En effet, pour un tel nœud, deux paramètres doivent être pris en compte dans le calcul de leur poids.

- Le premier paramètre est la distance du nœud par rapport au *NI* le plus proche, puisque l'objectif de la métrique est de router les flux sur un ensemble de nœuds d'intérêt. Pour y parvenir, il ne faut pas seulement réduire le poids de ces derniers mais aussi celui de leur voisinage afin de créer une force d'attraction vers eux. Aussi, plus un nœud est éloigné d'un *NI*, plus son poids doit être important. Ceci a pour conséquence de réduire son intérêt pour l'algorithme de routage.
- Le deuxième paramètre permet de différencier la centralité du nœud par rapport aux *NI*s. En effet, deux nœuds ayant une distance identique par rapport à leur *NI* le plus proche peuvent être, en réalité, placés à des endroits très différents du réseau (par exemple en périphérie ou au centre). C'est pourquoi le second paramètre représente le nombre de *NI*s se trouvant dans le voisinage d'un nœud. Plus un nœud a de *NI*s dans son voisinage et plus il y a de chance que ce nœud soit central par rapport à eux. Dans ce cas, son poids doit être plus faible afin d'en faire plus facilement un relai pour router des flux.

Ainsi, pour tout nœud n'étant pas un *NI*, son poids est égal à la distance le séparant du plus proche *NI*, divisée par le nombre de *NI*s se trouvant à cette distance. Plus précisément, le poids $FAME_v$ d'un nœud quelconque v n'appartenant pas à l'ensemble *NI* est le suivant :

$$FAME_v = \begin{cases} \frac{1}{|V|-1} & \text{si } v \in NI \\ \frac{d_{min}}{|NEI(v, d_{min}) \cap NI|} & \text{sinon} \end{cases}$$

avec d_{min} la distance séparant le nœud v de son plus proche *NI* et $NEI(v, d_{min})$ une fonction renvoyant le voisinage de v à la distance d_{min} .

4.1 Métrique d'agrégation de Flux : FAME

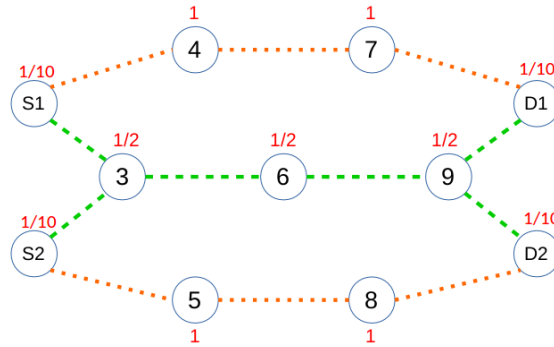


FIGURE 4.2 – Effet de FAME sur le routage.

La Figure 4.2 illustre ce principe en routant deux flux f_1 et f_2 , respectivement de S_1 à D_1 et de S_2 à D_2 , sur un réseau de onze nœuds. Dans cet exemple, il existe quatre NIs. Pour chaque nœud, son poids $FAME$ est précisé au-dessus. Si les flux sont routés sur le plus court chemin en termes de nombre de sauts (chemin orange), il est nécessaire d'utiliser huit nœuds (quatre NIs et quatre nœuds intermédiaires). En revanche, en utilisant la métrique FAME pour calculer la longueur des chemins, nous pouvons voir que le précédent routage n'est pas optimal (longueur de 2.1) et qu'il existe un meilleur, en vert (longueur 1.6) et n'utilisant que sept nœuds (quatre NIs et trois relais).

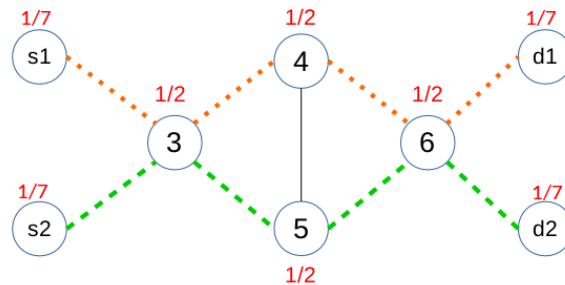


FIGURE 4.3 – Problème des réseaux connexes et symétriques.

La métrique FAME permet de router des flux vers un ensemble de nœuds d'intérêt et leurs voisinages. Ceci a pour effet de mettre en œuvre l'agrégation de flux. Toutefois, cette métrique dépend uniquement de la topologie du réseau, ce qui limite son efficacité dans les zones à forte densité ou symétriques. En effet, dans ces zones, des nœuds voisins peuvent avoir un poids identique, ce qui conduit potentiellement à l'apparition de chemins de même longueur. Dans une telle situation, l'algorithme de plus court chemin peut être amené à choisir un de ces chemins aléatoirement pour router un flux, ce qui a pour conséquence de réduire l'agrégation en utilisant éventuellement plus de nœuds que nécessaires.

La Figure 4.3 illustre ce principe. Les nœuds 4 et 5 ayant le même poids, les deux flux peuvent être routés en utilisant les huit nœuds. Pour résoudre ce problème, nous appliquons un facteur de réduction r ($0 < r < 1$) au poids des nœuds afin d'empêcher des voisins d'avoir le même poids. Plus précisément, le poids initial d'un nœuds v , calculé avec FAME ($FAME_v$), est multiplié par le facteur r afin de calculer le poids qui sera effectivement utilisé, noté W_v , par l'algorithme de routage. L'algorithme suivant montre comment la priorité est donnée aux nœuds en fonction de leur identifiant (adresse, numéro ou autre) :

Algorithm 2 Calcule de poids effectif des nœuds

```

1: procedure EFFECTIVEWEIGHTS
2:   for  $v \in V$  do ▷ Initialisation
3:      $W_v \leftarrow FAME_v$ 
4:   end for
5:   for  $v \in V$  do
6:     for  $n \in NEI(v, 1)$  do
7:       if  $W_v == W_n$  and  $v > n$  then
8:          $W_v \leftarrow W_v * r$  ▷ Le nœud v prend la priorité sur son voisin n.
9:       end if
10:    end for
11:  end for
12:  return  $W$  ▷ Renvoie les poids effectifs des nœuds.
13: end procedure

```

4.1.4 Résultats de simulation

Afin de rester cohérent dans les résultats présentés aux chapitre précédent, nous intégrons FAME dans l'algorithme de plus court chemin (en remplaçant la contrainte de capacité par la contrainte de clique) présenté page 29. Toutefois, la métrique FAME s'applique aux nœuds alors que la métrique de coût dans la modélisation concerne les liens du réseau. FAME représente le coût d'utilisation d'un nœud pour router un ensemble de flux. Nous pouvons donc facilement l'adapter à notre modélisation en remplaçant la fonction de coût $W_{i,j}$ d'un lien (i, j) par le poids effectif W_j calculé dans la partie précédente. De surcroit, la programmation linéaire nous permet de router les flux en prenant en compte les interférences, grâce à la contrainte de cliques, et donc de garantir la QoS en termes de débits requis.

Ces résultats nous permettent de comparer notre métrique d'agrégation FAME à notre solution optimale RA et au routage classique du plus court chemin en nombre de sauts. Nous détaillons également trois cas d'utilisation de FAME concernant le choix

4.1 Métrique d'agrégation de Flux : FAME

des nœuds d'intérêt. Le premier considère l'ensemble NI comme étant composé des nœuds sources et destinations et est dénommé FAME. FAME(random) considère que l'ensemble NI est obtenu par sélection aléatoire de nœuds. Enfin, FAME(cds) considère que NI est un ensemble dominant connecté du réseau.

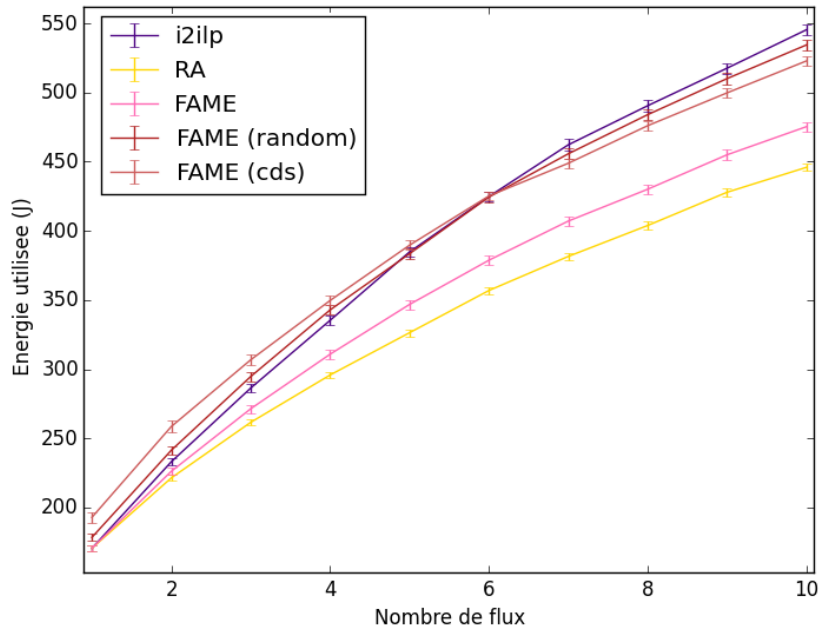


FIGURE 4.4 – Consommation énergétique globale en fonction du nombre de flux .

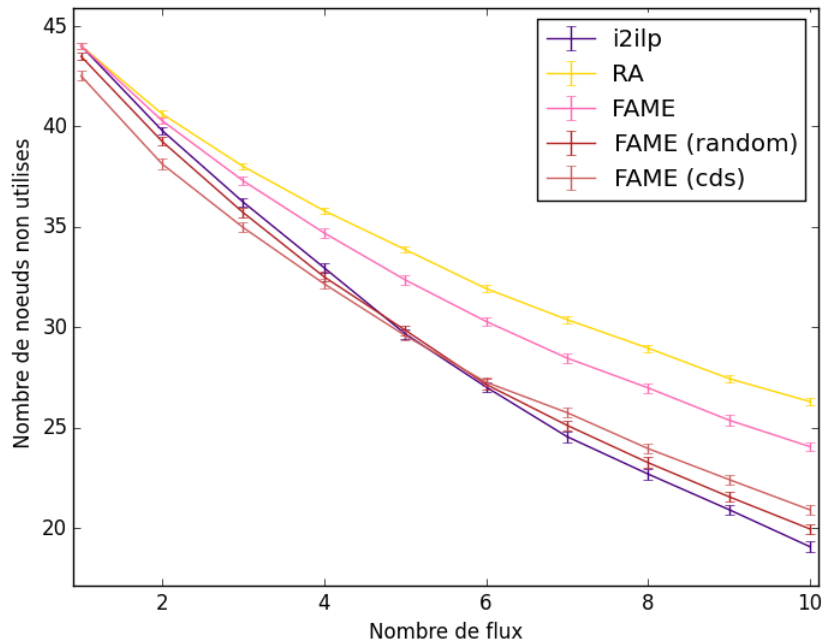


FIGURE 4.5 – Nombre de nœuds non utilisés en fonction du nombre de flux.

Les Figures 4.4 et 4.5 montrent la consommation énergétique globale du réseau et le nombre de nœuds pouvant être éteints en fonction du nombre de flux insérés dans le réseau. Nous pouvons constater que la métrique FAME est, bien que sous-optimale, très efficace en termes de consommation énergétique et d'agrégation. Nous remarquons également que l'efficacité de la métrique est fortement réduite lorsque l'ensemble des nœuds d'intérêt est construit aléatoirement ou à partir d'un CDS. Dans ces deux cas, l'efficacité énergétique et l'agrégation sont comparables au routage sur le plus court chemin en nombre de sauts. Il est donc clair que la façon de construire l'ensemble des nœuds d'intérêt a une influence sur l'efficacité de l'agrégation et qu'utiliser les nœuds sources et destinations est très efficace sur cette dernière. Nous verrons dans le prochain chapitre comment cette information peut être diffusée dans le réseau afin de l'utiliser dans un algorithme de routage distribué.

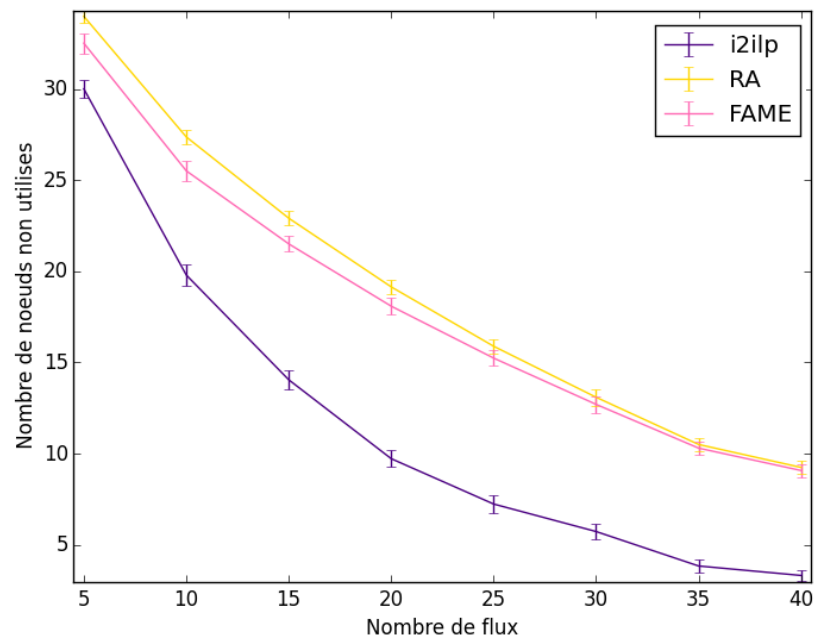


FIGURE 4.6 – Nombre de nœuds non utilisés en fonction du nombre de flux .

La Figure 4.6 montre l'efficacité de la métrique FAME lorsque le nombre de flux insérés dans le réseau est grand. Pour obtenir ces courbes, nous avons créé un autre ensemble de scénarii (c.f. Section 3.3.3) composé d'une unique topologie aléatoire où les flux ont chacun une demande en capacité minimale d'une unité. Nous pouvons voir qu'en plus d'être très proche de l'agrégation optimale, plus le nombre de flux dans le réseau augmente et plus l'agrégation avec la métrique FAME se rapproche de la solution optimale. Ceci s'explique par le fait que plus il y a de flux dans le réseau, plus l'ensemble des nœuds d'intérêt est grand. Il est donc plus facile d'agréger les flux sur ces nœuds puisqu'il y a moins de relais à utiliser pour relier les NIs.

4.1 Métrique d'agrégation de Flux : FAME

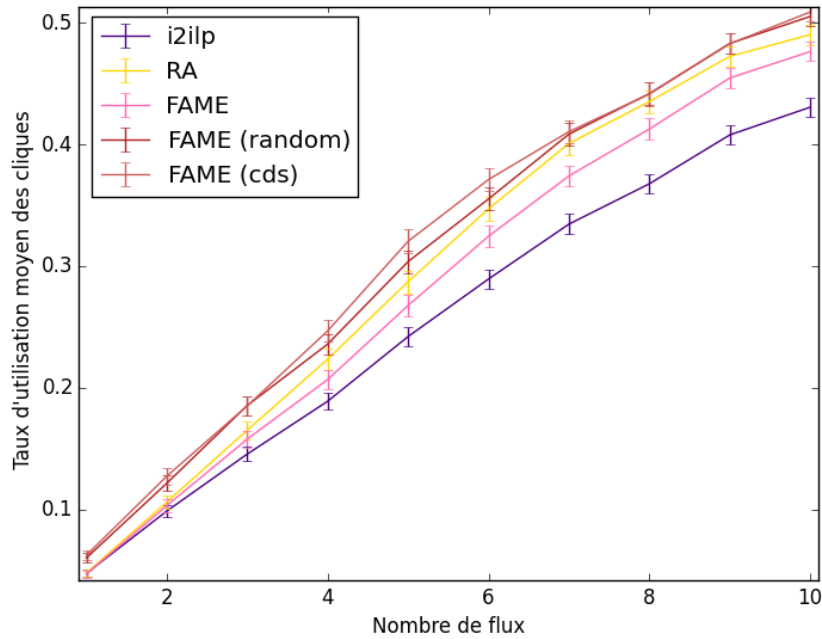


FIGURE 4.7 – Taux d'utilisation moyen des cliques en fonction du nombre de flux.

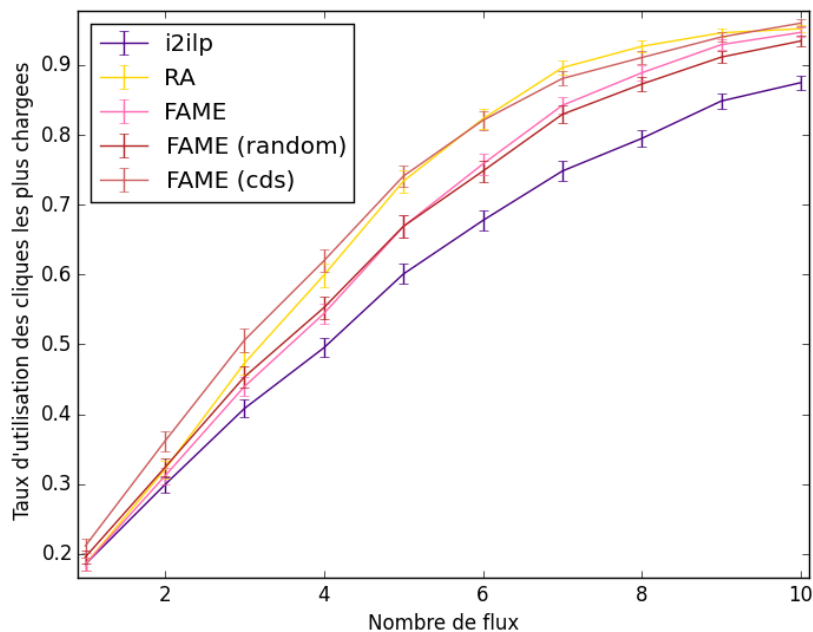


FIGURE 4.8 – Taux d'utilisation des cliques les plus chargées en fonction de nombre de flux.

La Figure 4.7 montre le taux d'utilisation moyen des cliques dans le réseau. Nous pouvons remarquer que, contrairement à ce que les précédents résultats auraient pu laisser penser, l'agrégation avec FAME charge moins le réseau en moyenne lorsque les nœuds d'intérêt sont les sources et les destinations que lorsqu'ils sont sélectionnés aléatoirement ou à partir d'un CDS. Cette tendance se retrouve sur la Figure 4.8 où

l'algorithme de routage utilisant FAME charge moins les cliques sur lesquelles les flux sont agrégés que la solution optimale, tout en étant très efficace en termes d'agrégation.

Nous pouvons donc conclure que FAME est une métrique de routage offrant une agrégation efficace aux algorithmes de plus court chemin. Cette efficacité nécessite toutefois de choisir judicieusement les nœuds d'intérêt. Bien que ce ne soit pas optimal, utiliser les nœuds sources et destinations permet d'avoir un bon compromis entre agrégation de flux et augmentation de la charge du réseau.

4.2 Prise en compte des interférences

Les résultats numériques précédents nous montrent l'efficacité de FAME lorsqu'elle est intégrée dans un algorithme de routage basé sur la programmation linéaire. Ceci nous permet de prendre en compte les interférences grâce à la contrainte de cliques présentée dans le Chapitre 3.13. Or cette formulation du routage sur le plus court chemin représente une solution centralisée et globale que nous ne souhaitons pas voir appliquée dans les réseaux sans fil multi-sauts. De plus, la Figure 4.9 montre que lorsque FAME est intégrée dans un algorithme de plus court chemin ne prenant pas en compte les interférences, les cliques du réseau peuvent être surchargées, c'est-à-dire dépasser les 100% de taux d'utilisation. Il est donc primordial d'insérer des mécanismes de sécurité permettant de limiter la surcharge des zones du réseau où l'agrégation s'opère.

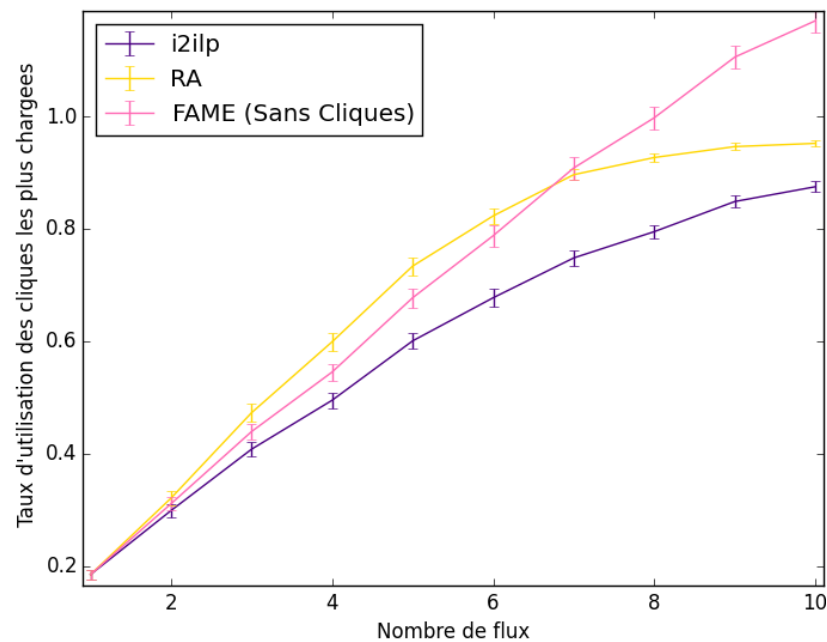


FIGURE 4.9 – Taux d'utilisation des cliques les plus chargées en fonction de nombre de flux.

4.2 Prise en compte des interférences

Pour pouvoir prendre en compte les interférences avec FAME, deux approches sont possibles. La première est celle que nous avons utilisée dans la partie précédente et consiste à intégrer la métrique dans un algorithme de routage qui gère les interférences. Cette approche possède toutefois le défaut de lier l'utilisation de FAME à un algorithme (ou type d'algorithme) en particulier, ce qui restreint l'ambition initiale de pouvoir proposer une solution générique. La deuxième approche consiste à intégrer la prise en compte des interférences dans le calcul de la métrique. Ceci a pour avantage de la rendre indépendante de l'algorithme de routage avec lequel elle est utilisée et donc beaucoup plus facilement intégrable dans les algorithmes de routage existants.

4.2.1 Le problème des cliques

Jusqu'à présent, les interférences étaient prises en compte grâce à un modèle d'interférences à base de cliques (cf. Section 3.1.2). Or le calcul du graphe de conflits et des cliques sur ce dernier voit sa complexité augmenter exponentiellement en fonction de la taille du réseau considéré. De plus, ce calcul nécessite que chaque nœud puisse disposer d'une vision globale du réseau (topologie et taux d'utilisation des liens). Si la topologie est déjà une information nécessaire pour le calcul de FAME et peut être facilement diffusée grâce à un protocole de découverte de topologie ou de routage proactif (par exemple OLSR [16, 24]), cela est plus délicat concernant le taux d'utilisation des liens. En effet, le nombre de liens dans le réseau peut être très grand. C'est pourquoi diffuser périodiquement les données les concernant peut demander l'usage d'une grande partie de la capacité du réseau, au point de réduire significativement celle disponible pour les utilisateurs. Par ailleurs, bien qu'il existe des protocoles intégrant la diffusion de métriques de liens tels que OLSRv2 [24], il reste le problème de la complexité du calcul des cliques.

4.2.2 Approximation des cliques

Pour résoudre ce problème d'interférences, nous approximations le modèle de cliques en considérant les interférences entre les nœuds plutôt qu'entre les liens et en réduisant le calcul des cliques à un calcul de voisinage. Ce nouveau modèle se base sur les propriétés des transmissions sans fil suivantes. La première est que la réception de données est une action passive qui ne génère pas d'interférences. Par conséquent, plusieurs nœuds peuvent être en état de réception en même temps. Inversement, la transmission de données est une action active générant des interférences. Plusieurs nœuds interférant entre eux ne peuvent pas transmettre des données en même temps.

Ceci nous amène à considérer le taux de transmission d'un nœud, à savoir le ratio de son débit de transmission sur la capacité de transmission du support physique (qui peut être déterminé par les spécifications de la technologie utilisée), comme la portion de temps que le nœud passe à utiliser le support de communication. Ainsi, en s'inspirant du modèle de cliques, nous construisons un ensemble interférant pour chaque nœud dont nous calculons le taux de transmission comme étant la somme des taux de transmission des nœuds composant cet ensemble. Enfin, comme nous considérons un modèle d'interférences à deux sauts, nous construisons l'ensemble interférant d'un nœud à partir de son voisinage à deux sauts. Nous pouvons donc calculer la charge du voisinage d'un nœud, notée NL (Neighborhood Load), comme suit :

$$\forall v \in V, NL_v = \sum_{i \in NEI(v,2)} \sum_{(i,j) \in E} \frac{d_{i,j}}{C_{i,j}}$$

En suivant le même schéma que celui utilisé pour la construction de la contrainte des cliques, nous pouvons considérer qu'un ensemble de flux est réalisable dans le réseau si la charge du voisinage de chaque nœud du réseau est inférieure à 100%. Toutefois, cette nouvelle formule n'a pas été développée pour élaborer une nouvelle contrainte mais pour mettre au point un mécanisme de sécurité dans le calcul de FAME et ainsi éviter une agrégation excessive des flux.

4.2.3 Métrique d'agrégation de flux adaptative : A-FAME

Le principe de A-FAME (Adaptative Flow Aggregation Metric) consiste à réduire la force d'agrégation de la métrique FAME au fur et à mesure que la charge de réseau augmente. Nous avons pu observer qu'agrégérer des flux augmente sensiblement la charge du réseau. Ceci peut avoir des conséquences négatives sur la capacité du réseau à supporter de nouveaux flux ou à répondre aux demandes des flux existants.

Pour résoudre ce problème, nous proposons une solution consistant à agréger les flux en suivant la métrique FAME lorsque le réseau n'est pas très chargé, puis à alléger l'agrégation lorsque la charge de réseau atteint un certain seuil noté Th . Ce comportement peut être obtenu en normalisant le poids des nœuds autour d'une moyenne. En effet, plus le poids des nœuds est proche de cette moyenne, plus l'algorithme de routage va tendre à router les flux sur le chemin le plus court. Inversement, plus le poids des nœuds est proche de la valeur calculée par FAME, plus le routage va agréger les flux. La métrique A-FAME nécessite donc un calcul supplémentaire, en se basant sur la charge actuelle du réseau.

4.2 Prise en compte des interférences

Elle est calculée pour chaque nœud de la manière suivante :

$$AFAME_v = \alpha_v \cdot FAME_v + (1 - \alpha_v) \frac{1}{|V|} \sum_{j \in V} FAME_j$$

avec α_v la force d'agrégation de la métrique A-FAME sur le nœud v , comprise entre 0 et 1. Plus α_v tend vers 1, plus A-FAME est proche de la valeur calculée par FAME. En revanche, si α_v tend vers 0, l'agrégation des flux sur le nœud v est nulle puisque son poids est égal à la moyenne des poids FAME de tous les nœuds du réseau. Le nœud v n'est donc ni favorisé ni évité par l'algorithme de routage.

Hyperparamètre : seuil de réduction et base d'agrégation

Pour pouvoir appliquer la métrique A-FAME, deux paramètres doivent être déterminés :

- 1) Le seuil de réduction, noté Sr , à partir duquel nous considérons que le voisinage d'un nœud est trop chargé et qu'il faut donc l'éviter pour le routage.
- 2) La base d'agrégation, noté Ba , représentant pour un nœud la valeur initiale de sa force d'agrégation lorsque le seuil Sr est dépassé.

Plus précisément, pour un nœud quelconque v , la valeur de sa force d'agrégation α_v est fonction de Sr et Ba , ainsi que de la charge de son voisinage NL_v . Elle est calculée comme suit :

$$\alpha_v = \begin{cases} 1 & \text{si } NL_v \leq Sr \\ \max(0, Ba - (NL_v - Sr)) & \text{sinon} \end{cases}, \forall v \in V$$

Déterminer les valeurs optimales à donner aux deux hyperparamètres que sont Sr et Ba est un problème d'optimisation difficile. Un seuil de réduction trop élevé risque de ne pas permettre la détection d'une surcharge du réseau et donc d'avoir une utilisation de la métrique A-FAME inefficace. En effet, plus le seuil est élevé, plus l'impact de la surcharge sur la force d'agrégation est faible. En revanche, si le seuil est trop bas, l'agrégation des flux sera réduite, à cause d'une utilisation excessive de A-FAME. Quant à la base d'agrégation, une valeur élevée a pour effet de limiter l'amplitude possible de la force d'agrégation, annihilant l'adaptabilité de la métrique. Inversement, une valeur faible entraîne une réduction excessive de l'agrégation et peut conduire à un phénomène d'oscillation où, au moment de chaque mise à jour de l'algorithme de routage, les métriques (FAME et A-FAME) sont utilisées alternativement. Nous montrons dans la sous-section suivante l'effet de ces paramètres sur l'efficacité de notre contribution.

4.2.4 Résultats de simulation

Les simulations suivantes visent à comparer l'efficacité de la métrique d'agrégation de flux adaptative A-FAME avec la métrique présentée en première partie de ce chapitre, FAME. A-FAME est intégrée dans un algorithme de plus court chemin, sans prise en compte des interférences, toujours implémenté avec la programmation linéaire en nombres entiers pour des raisons de cohérence avec les solutions de référence. Nous évaluons l'influence des hyperparamètres appliqués à A-FAME ($Ba - Sr$), où Ba est la base d'agrégation et Sr le seuil de réduction que nous fixons arbitrairement à une valeur élevée (80%), puis à une valeur faible (50%).

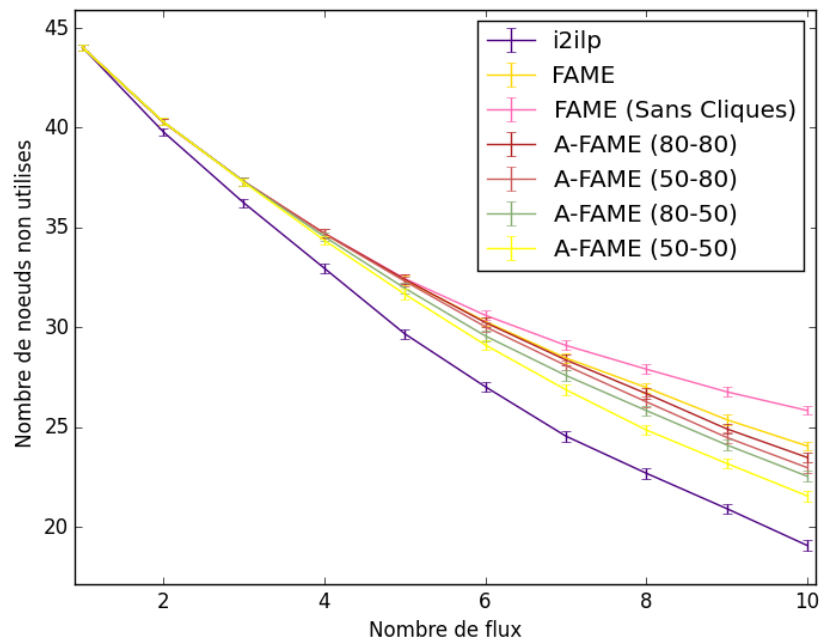


FIGURE 4.10 – Nombre de nœuds non utilisés en fonction du nombre de flux.

La Figure 4.10 montre le nombre de nœuds non utilisés, pouvant alors être éteints, en fonction du nombre de flux insérés dans le réseau. Nous pouvons voir l'influence des hyperparamètres sur la capacité de la métrique A-FAME à agréger les flux. Comme vu précédemment, lorsque la métrique FAME est intégrée à l'algorithme de routage en ignorant la contrainte de cliques, le niveau d'agrégation est meilleur mais au prix de voir la capacité du réseau insuffisante pour acheminer tous les flux, engendrant une baisse de la QoS. A-FAME permet de résoudre ce problème en intégrant la prise en compte des interférences dans le calcul de la métrique plutôt que dans l'algorithme de routage. Nous pouvons constater que lorsque A-FAME est utilisée avec une base d'agrégation et un seuil de réduction élevés (80%), nous nous retrouvons avec un niveau d'agrégation très proche de celui de FAME. Ceci est dû aux valeurs importantes des hyperparamètres, qui évitent de réduire significativement l'agrégation de flux tout

4.2 Prise en compte des interférences

en rendant les conditions de cette réduction plus difficiles à remplir. Enfin, nous pouvons remarquer que lorsque le seuil de réduction est bas (50%), l'agrégation est moins efficace. Ceci peut s'expliquer par le fait qu'avec un tel seuil de réduction, la métrique A-FAME soit plus sensible à la charge du réseau et tende donc à réduire davantage et plus souvent la force d'agrégation.

Les Figures 4.11 et 4.12 montrent les taux d'utilisation, moyen et maximal, des cliques en fonction du nombre de flux et permettent ainsi d'évaluer l'effet de la métrique A-FAME sur la charge du réseau. Nous pouvons remarquer que notre métrique permet de réduire la charge moyenne du réseau de manière significative (réduction de plus de 50% de l'écart avec le plus court chemin) et ce, malgré l'utilisation des meilleurs paramètres en termes d'agrégation (80-80).

Enfin, la Figure 4.12 montre que lorsque peu de flux sont insérés et que le réseau est peu chargé, le routage selon la métrique A-FAME suit le routage avec FAME, jusqu'à un certain niveau de charge. Une fois ce dernier atteint, qui dépend du seuil de réduction utilisé, A-FAME commence à s'écarter de FAME pour se rapprocher du comportement de l'algorithme de routage sur le plus court chemin. Par ailleurs, nous pouvons voir que la courbe A-FAME (80-80) suit de très près celle de FAME pour l'ensemble des scénarii que nous utilisons depuis le début de ce document (cinq topologies aléatoires avec des flux aux demandes variables). Ces valeurs semblent donc être pertinentes pour une mise en application de la métrique d'agrégation avec prise en compte des interférences.

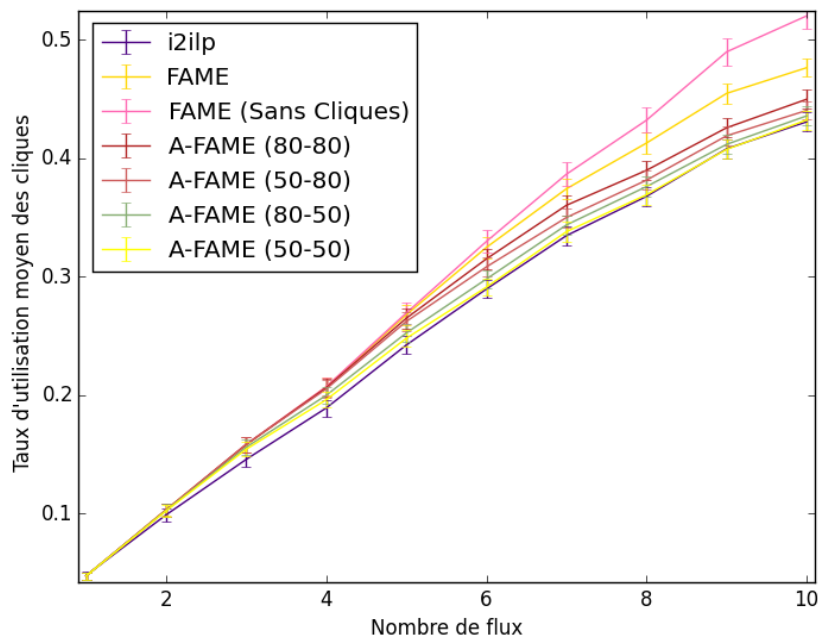


FIGURE 4.11 – Taux d'utilisation moyen des cliques en fonction du nombre de flux .

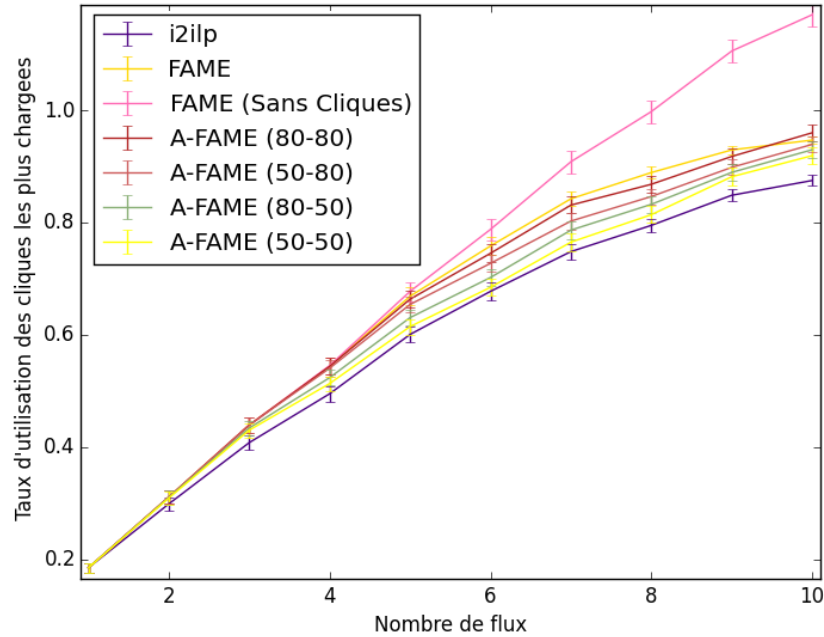


FIGURE 4.12 – Taux d'utilisation des cliques les plus chargées en fonction de nombre de flux.

4.3 Conclusion

Dans ce chapitre, nous avons proposé une solution permettant d'agréger un ensemble de flux sur un minimum de nœuds, sans avoir à passer par la programmation mathématique dont la résolution à l'optimum peut être difficile à obtenir. La première solution repose sur l'utilisation d'une métrique de routage permettant aux algorithmes de routage de plus court chemin de router les flux du réseau en priorité sur un ensemble de nœuds identifiés comme des nœuds d'intérêt. Cette approche est efficace en termes d'agrégation, particulièrement lorsque le nombre de flux insérés est important, mais nécessite l'emploi d'un algorithme de routage pouvant prendre en compte les interférences afin d'éviter de surcharger les zones du réseau où les flux sont agrégés.

Pour résoudre ce problème, une seconde solution a été proposée, à savoir appliquer à la première métrique un facteur dépendant de la charge du réseau, afin de limiter l'agrégation des flux lorsque cette dernière est trop importante. Cette solution permet ainsi d'appliquer l'agrégation de flux avec des algorithmes de routage de plus court chemin ne prenant pas en compte les interférences et est donc, potentiellement, plus facilement intégrable dans les protocoles de routage existants.

4.3 Conclusion

Pour pouvoir appliquer l'agrégation de flux, quelques informations sont requises pour que les nœuds puissent calculer leur table de routage. Dans le contexte de la première solution, seules celles concernant l'appartenance d'un nœud à l'ensemble des nœuds d'intérêt (*NI*s), les nœuds sur lesquels l'algorithme de routage doit diriger les flux, la distance séparant ces nœuds d'un *NI* et la connaissance d'une partie de leur voisinage sont nécessaires.

Les deux premières informations peuvent être facilement obtenues au moyen d'un processus de diffusion d'un bit de donnée, avec mémorisation de la distance parcourue, en nombre de sauts, par ce dernier. Le voisinage d'un nœud, en revanche, peut nécessiter l'utilisation d'un protocole de découverte de topologie plus coûteux en ressource réseau. Dans le cas de l'application de la seconde solution, un nœud doit également diffuser son taux de transmission (pourcentage de temps passé en mode transmission), afin de permettre aux autres nœuds de calculer la charge de son voisinage. Cette information, majorée à 100%, peut être codée sur quelques bits uniquement, ce qui ramène le coût total d'intégration de la seconde solution à un octet maximum, à diffuser sur l'ensemble du réseau pour chaque nœud. Cela représente une augmentation très négligeable par rapport à la capacité nécessaire au fonctionnement d'un protocole de routage proactif tel que OLSR [16].

Le prochain chapitre abordera donc la problématique de la mise en application de l'agrégation de flux, dont une partie sera consacrée à l'intégration de la métrique FAME dans le protocole de routage OLSR avec le simulateur de réseau NS-3 [62].

Chapitre 5

Application

Nous avons proposé dans le précédent chapitre une métrique efficace permettant à des algorithmes de routage de plus court chemin d'agrèger un ensemble de flux sur un nombre réduit de nœuds, sans remettre en cause la qualité de service en termes de débit. Une extension de cette métrique a également été proposée, afin de permettre la gestion des interférences via une approximation de la contrainte des cliques. Les informations nécessaires pour les nœuds utilisant ces métriques ont été décrites, leur permettant de calculer leur table de routage.

L'intégration de ces métriques d'agrégation passe par l'implémentation d'algorithmes distribués de diffusion d'informations et de découverte de topologie. Nous avons rappelé dans le Chapitre 2 les protocoles de routage intégrant ces fonctionnalités. Notamment, OLSR (c.f. Section 2.1.1) est un protocole de routage proactif permettant aux nœuds d'un réseau de découvrir la topologie de ce dernier grâce à la diffusion de messages spécifiques, à intervalle régulier. Ces messages permettent à chacun des nœuds de reconstruire une topologie partielle du réseau à partir de laquelle ils vont pouvoir calculer leur table de routage. Il est donc possible d'ajouter à ces messages les informations manquantes pour permettre le calcul de la métrique d'agrégation et de la table de routage des nœuds.

La mise en application du routage avec agrégation de flux reste toutefois sujette à un certain nombre de questions. La potentielle surcharge locale qui est créée sur les chemins où les flux sont agrégés peut engendrer une dégradation de l'expérience utilisateur, surtout de nos jours où il est courant de faire transiter sur les réseaux du trafic important tel que de la vidéo. Nous avons montré qu'il était possible d'associer des stratégies pour améliorer les performances générales du réseau ou de l'agrégation, telles que le codage réseau (cf. Section 3.5) ou la définition des nœuds d'intérêt

5.1 Rappel sur le protocole de routage OLSR

(cf. Section 4.1.2). Mais il reste la question de la technique d'économie d'énergie. En elle-même, l'agrégation de flux n'offre aucune réduction de la consommation énergétique du réseau. Les gains énergétiques montrés dans les chapitres précédents sont obtenus lorsqu'une méthode d'extinction des nœuds est utilisée.

La première partie de ce chapitre complète donc d'une manière plus technique le fonctionnement du protocole OLSR, déjà abordé dans la Section 2.1.1. Plus précisément, nous présentons les messages utilisés par le protocole que nous adapterons afin d'y ajouter les informations nécessaires à l'intégration de notre métrique FAME dans le protocole de routage OLSR. Ces modifications feront l'objet de la partie suivante dans laquelle nous introduisons notre version du protocole OLSR avec agrégation de flux. Une troisième partie présente les résultats obtenus grâce à un jeu de simulations exécutées dans l'environnement du simulateur de réseau NS-3 [62]. Enfin, une conclusion résume les points marquants de ce chapitre et discute des perspectives qu'ouvre le routage avec agrégation de flux, qui seront plus approfondies dans le dernier chapitre de cette thèse.

5.1 Rappel sur le protocole de routage OLSR

Le protocole OLSR est un protocole de routage proactif (c.f. Section 2.1.2) implémentant un protocole de découverte de topologie, auquel vient s'ajouter un algorithme de routage de plus court chemin en nombre de sauts. Ce protocole repose principalement sur la diffusion régulière de deux types de messages.

Les messages HELLO sont envoyés par les nœuds afin d'annoncer leur présence à leurs voisins. La Figure 5.1 montre la structure d'un tel message et les champs qui y sont associés :

- "Reserved" est un champ réservé qui n'est pas utilisé par le protocole. Une implémentation du protocole doit fixer ce champ à 0 afin de correspondre à la spécification [16] standard.
- "Htime", sur un octet, représente l'intervalle de temps séparant l'envoi de deux messages Hello consécutifs.
- "Willingness" informe la propension du nœud à l'origine de ce message à prendre le rôle de MPR (servir de relai pour les phases de diffusion et de retransmission des paquets).

- "Link Code" spécifie le type (symétrique, asymétrique, perdu, état inconnu) des liens entre l'émetteur du message et les voisins annoncés dans la liste d'adresses fournie dans ce message.
- "Reserved" est un autre champ réservé, de un octet, qui doit également être fixé à 0 afin d'être conforme à la spécification.
- "Link Message Size" représente la taille, en octets, de la liste de voisins qui suit.
- "Neighbor Interface Address", sur quatre octets, code la liste des adresses des voisins que le nœud émetteur annonce.
- La liste précédente peut être suivie par un nouveau bloc de données recommençant par le champ "Link Code", afin d'annoncer une autre liste de liens ayant un type différent.

0								1								3								4											
0	1	2	3	4	5	6	7	8	0	1	2	3	4	5	6	7	8	0	1	2	3	4	5	6	7	8	0	1	2	3	4	5	6	7	8
Reserved								Htime								Willingness																			
Link Code								Reserved								Link Message Size																			
Neighbor Interface Address																																			
Neighbor Interface Address																																			
...																																			
Reserved								Htime								Willingness																			
Link Code								Reserved								Link Message Size																			
Neighbor Interface Address																																			
Neighbor Interface Address																																			
...																																			

FIGURE 5.1 – Structure d'un message HELLO.

Ces messages Hello permettent aux nœuds les recevant de construire une topologie locale de leur voisinage à deux sauts. Ils connaissent aussi la qualité des liens qu'ils ont avec leurs voisins et ceux entre ces derniers et leurs propres voisins.

La découverte du reste de la topologie est assurée par la diffusion des messages TC (Topology Control). Ces messages sont diffusés par les MPRs qui annoncent, par ce biais, leurs voisins avec lesquels ils disposent d'un lien symétrique. La Figure 5.2 illustre la structure des messages TC. Les deux premiers champs sont un numéro de séquence (ANSN), permettant d'identifier et de dater le message sur deux octets, et un espace réservé (Reserved) de deux octets devant être fixés à zéro afin que l'implémentation du protocole soit conforme à la spécification. Ensuite, une liste de champs "Advertise Neighbor Main Address" permet au nœud émetteur de lister ses voisins.

5.2 OLSR avec agrégation de flux (A-OLSR)

0								1								2								3											
0	1	2	3	4	5	6	7	8	0	1	2	3	4	5	6	7	8	0	1	2	3	4	5	6	7	8	0	1	2	3	4	5	6	7	8
ANSN																Address Count																			
Advertized Neighbor Main Address																																			
Advertized Neighbor Main Address																																			
...																																			

FIGURE 5.2 – Structure d'un message TC (Topology Control).

Grâce à cette fonctionnalité de découverte de topologie et au calcul des routes sur le plus court chemin, le protocole de routage OLSR est un candidat intéressant pour appliquer la métrique FAME. De plus, ce protocole a été conçu afin de pouvoir intégrer des extensions. Il est donc possible de construire un nouveau type de message sous la forme d'une extension dans laquelle nous pouvons diffuser les informations manquantes aux nœuds pour pouvoir calculer la métrique FAME puis les routes sur le plus court chemin, en fonction de cette dernière. Il est important de souligner que la nouvelle mouture du protocole, OLSRv2 [24], datant de 2014, intègre par défaut le routage sur le plus court chemin avec métrique (la première version ne faisant que du plus court chemin en nombre de sauts), ainsi que la diffusion de métrique de liens permettant ainsi une intégration de FAME, plus facile, sans avoir à développer une extension particulière.

5.2 OLSR avec agrégation de flux (A-OLSR)

La première modification à faire dans le protocole OLSR pour y intégrer la métrique FAME est de permettre à un nœud de notifier son statut (NI ou non) à ses voisins. Pour ce faire, nous utilisons un des octets du premier champ "Reserved" du message HELLO, le premier bit permettant de préciser si le nœud émetteur du message est un nœud d'intérêt (valeur 1) ou non (valeur 0).

Dans le cas où nous voudrions intégrer la version adaptative de FAME, les sept autres bits de l'octet "Reserved" utilisé conviendraient pour diffuser le taux de transmission (rapport entre le débit et la capacité du lien) du nœud émetteur sous la forme d'un pourcentage. Ainsi, sur réception d'un message HELLO, un nœud peut évaluer ce premier champ et identifier si le nœud émetteur appartient à l'ensemble des *NI*s. Cette information est ensuite sauvegardée dans une table indexée par les adresses des nœuds connus, et reste valide jusqu'à la mise à jour de cette valeur (sur réception d'un nouveau message HELLO ou TC) ou la fin de sa période de validité (paramétrée lors du déploiement du réseau).

Pour qu'un nœud puisse détecter s'il est NI (source ou destination), une vérification lors de l'émission et de la réception d'un paquet peut être faite. En effet, lors de l'envoi d'un paquet, si son adresse d'origine est le nœud faisant cette vérification, alors ce dernier est l'émetteur de ce paquet et donc la source du flux auquel il appartient. Ce nœud peut se déclarer NI. De même, lors de la réception d'un paquet, si l'adresse destinataire est le nœud faisant la vérification, alors ce dernier est la destination du flux auquel appartient ce paquet. Ce nœud peut également se déclarer NI. Si aucune des deux situations ne se présente, alors le nœud n'est qu'un relai pour le flux considéré.

Les modifications précédentes, illustrées par la Figure 5.3a, permettent aux nœuds d'informer leur voisins de leur statut. Pour pouvoir diffuser cette informations au-delà de ce voisinage, il est nécessaire de modifier les messages TC. De nouveau, nous utilisons le champ "Reserved" du message pour y renseigner le nombre d'adresses qui sont annoncées par le nœud émetteur. Ceci nous permet de connaître la taille de la liste d'adresses qui suit ce champ. Par conséquent, nous pouvons ajouter au message, à la suite de cette liste, une seconde liste d'octets contenant le statut des nœuds annoncés. Cette dernière respecte le même ordre que la liste d'adresses qui la précède, de sorte que le i -ème élément de la liste des statuts corresponde au i -ème nœud annoncé par la liste d'adresses (cf. Figure 5.3b. Ainsi, sur réception d'un message TC, un nœud peut compléter ou mettre à jour sa table des nœuds d'intérêt.

Enfin, l'algorithme de routage du plus court chemin de OLSR doit être modifié pour prendre en compte ces nouvelles informations. Premièrement, le nœud doit calculer la topologie du réseau en fonction des informations recueillies grâce au fonctionnement de base du protocole OLSR. Ensuite, le calcul de la métrique peut être fait selon la méthode décrite au Chapitre 4. Une modification de l'algorithme de plus court chemin doit enfin être faite pour calculer la longueur des routes selon la métrique FAME (et non plus en fonction du nombre de sauts).

5.3 Simulations

0									1									2									3								
0	1	2	3	4	5	6	7	8	0	1	2	3	4	5	6	7	8	0	1	2	3	4	5	6	7	8	0	1	2	3	4	5	6	7	8
Node state									Reserved									Htime									Willingness								
Link Code									Reserved									Link Message Size																	
Neighbor Interface Address																																			
Neighbor Interface Address																																			
...																																			
Node state									Reserved									Htime									Willingness								
Link Code									Reserved									Link Message Size																	
Neighbor Interface Address																																			
Neighbor Interface Address																																			
...																																			

(a) Message HELLO

0									1									2									3								
0	1	2	3	4	5	6	7	8	0	1	2	3	4	5	6	7	8	0	1	2	3	4	5	6	7	8	0	1	2	3	4	5	6	7	8
ANSN																		Address Count																	
Advertized Neighbor Main Address																																			
...																																			
Advertized Neighbor Main Address																																			
Node state									Node state																
...																

(b) Message TC

FIGURE 5.3 – Structures des messages HELLO et TC après intégration de FAME.

5.3 Simulations

5.3.1 Network Simulator 3

Le simulateur open source NS-3 [62] est un simulateur de réseau à événements discrets, codé en C++ et dont l'usage est principalement dédié à l'enseignement et à la recherche. Il est le fruit d'une collaboration entre l'INRIA et l'université de Washington, auxquels se sont par la suite associées d'autres institutions. Il intègre la plupart des fonctionnalités de base des réseaux (IP et non-IP), ainsi que des modèles de transmission pour des réseaux sans fil de différents types (par exemple Wi-Fi ou LTE). Il intègre également par défaut, et c'est l'une des principales raisons qui nous a poussés à choisir cet outil, une implémentation du protocole OLSR que nous avons pu réutiliser et étendre avec notre métrique d'agrégation, ainsi que le même modèle énergétique à états que nous utilisons dans le Chapitre 3.

La méthode de travail classique avec ce simulateur consiste à télécharger le projet complet, à en modifier les sources en fonction de nos besoins (comme tester une nouvelle fonctionnalité dans un algorithme existant pour notre cas), recompiler le simulateur

en entier avec notre simulation et exécuter cette dernière. Cette approche de travail, bien que fonctionnelle, fiable et recommandée par les personnes qui maintiennent le projet, s'avère être lourde à mettre en pratique et plutôt inconfortable. C'est pourquoi nous avons compilé une unique fois le projet afin de lier ses bibliothèques de manière classique à nos programmes de simulation. Nous avons dupliqué le code du protocole OLSR afin de le transformer en un A-OLSR (Agregated-OLSR), en intégrant notre métrique FAME. L'algorithme de routage a bien sûr été modifié et suit désormais les étapes suivantes :

1. Construction de la topologie partielle du réseau.
2. Calcul de la métrique FAME pour les nœuds contenus dans cette topologie.
3. Exécution de l'algorithme de Dijkstra (plus court chemin) en fonction de FAME.
4. Mise à jour de la table de routage du nœud.

Enfin, une interface avec nos précédents outils (c.f. Section 3.3.3) a été développée afin de permettre à nos simulations sous NS-3 d'utiliser des graphes préconstruits, de router le trafic de façon statique selon le résultat d'une solution issue de la résolution d'un programme d'optimisation linéaire, ou d'utiliser les paramètres énergétiques propres à un graphe considéré. Toutefois, cette interaction reste assez limitée étant donné que certains paramètres ne peuvent pas être fixés arbitrairement dans NS-3. C'est notamment le cas pour la capacité des liens qui est généralement propre au modèle physique utilisé. Pour cette raison, les résultats présentés ci-après ont été obtenus à partir d'un nouvel ensemble de scénarii dont nous détaillons les paramètres de création.

5.3.2 Simulations et résultats numériques

Dans ce chapitre, nous réalisons nos simulations au moyen du simulateur de réseau NS-3. Comme expliqué ci-dessus, nous devons générer de nouveaux scénarii afin que ces derniers puissent être compatibles avec les fonctionnalités du simulateur. Les principaux paramètres à adapter pour permettre l'utilisation de nos scénarii aléatoires sont la capacité des liens (que nous fixons désormais à 1 Mbps), le temps d'arrivée des flux dans le réseau et la durée de ces derniers. Pour des raisons de simplicité et parce que nous avons vu dans les chapitres précédents que l'efficacité de l'agrégation de flux dépendait du nombre de flux présents dans le réseau, nous faisons en sorte que les flux commencent tous au même instant (après 30 secondes) et restent dans le réseau pendant 30 secondes. Ces flux sont simulés par un modèle à débit constant (CBR) de 10 Kbps.

5.3 Simulations

L'utilisation d'un débit faible nous permet d'insérer un grand nombre de flux dans le réseau, ce qui est l'un des facteurs importants influençant le nombre de nœuds utilisés. Nous pouvons donc créer des simulations dans lesquelles le nombre de flux présents dans le réseau varie à un instant donné. Le fait de ne faire entrer les flux qu'au bout de 30 secondes permet de laisser le temps au protocole OLSR de converger et d'offrir une route à chacun lors de leur insertion. Enfin, nous générons 50 scénarii pour un nombre donné de flux, sur une unique topologie, et ce pour des raisons de complexité et de temps de simulation.

Nouvelles mesures

L'utilisation d'un simulateur de réseau nous permet d'observer de nouvelles mesures, en rapport avec les performances du réseau. Ainsi, nous pouvons désormais mesurer le taux de paquets perdus de bout-en-bout, le taux de paquets rejetés au niveau de la couche MAC, ou encore la consommation énergétique du réseau. Pour cette dernière, nous considérons que la consommation énergétique d'un nœud nécessaire pour sa mise sous tension est de 24 Joules (12 Volts et 2 Ampères). Par ailleurs, l'aspect temporel étant pris en compte dans les simulations, nous devons redéfinir notre manière d'évaluer le nombre de nœuds utilisés dans le réseau. Pour ce faire, nous intégrons un déclencheur dans le protocole de routage afin que ce dernier garde une trace datée de chaque moment où un nœud a eu à router un paquet d'un flux. La période de temps couvrant ces traces est ensuite discrétisée en intervalles de deux secondes et la moyenne du nombre de nœuds utilisés dans chacun de ces intervalles nous donne le nombre moyen de nœuds utilisés durant la simulation d'un scénario.

Résultats

Ces résultats visent à évaluer l'applicabilité de la métrique FAME et les effets de son intégration dans un protocole de routage proactif. Les Figures 5.4 et 5.5 montrent l'efficacité de FAME en termes d'agrégation et son impact énergétique, et comparent son intégration dans OLSR (A-OLSR) à une méthode de routage statique générée par l'optimisation linéaire. Les figures suivantes montrent l'impact de FAME sur la consommation énergétique et les performances du réseau lorsque une méthode d'extinction des nœuds est utilisée par dessus le routage, afin de réduire la consommation énergétique globale. Nous mesurons la consommation énergétique pendant la période de temps où les flux sont présents dans le réseau. Par conséquent, c'est également au début de cette période que les nœuds non utilisés sont éteints. Ceci permet d'éviter les problèmes posés par l'extinction des nœuds dans les protocoles de routage proactifs en général.

En effet, éteindre des nœuds provoque des changements dans la topologie du réseau, ce qui affecte à la fois le fonctionnement du protocole et la stabilité des routes calculées. Le développement d'un algorithme d'extinction de nœuds compatible avec un tel protocole de routage est un travail difficile et dans lequel de nombreux paramètres doivent être pris en compte, tels que les différents intervalles de transmission, les lois de probabilités liées à l'arrivée des flux, à l'état des nœuds, éventuellement la mobilité ou encore le temps pendant lequel un nœud reste éteint. Un tel travail de développement sort du cadre de cette thèse mais reste un sujet qui devrait être étendu en tant que perspective.

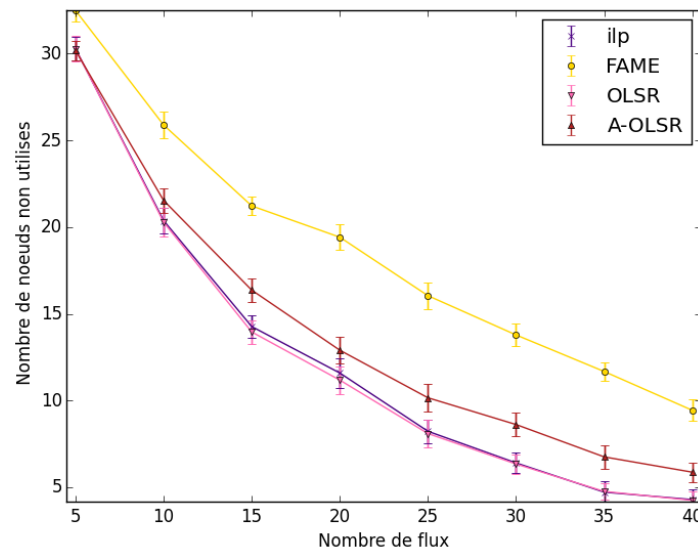


FIGURE 5.4 – Nombre moyen de nœuds utilisés en fonction du nombre de flux.

La Figure 5.4 montre le nombre de nœuds non utilisés en fonction du nombre de flux insérés dans le réseau. Nous pouvons voir que notre métrique d'agrégation permet d'augmenter le nombre moyen de nœuds non utilisés dans le réseau lorsqu'elle est intégrée à OLSR. Ceci confirme l'applicabilité de notre solution. Cependant, nous pouvons remarquer que l'efficacité de FAME est bien inférieure lorsqu'elle est employée dans le protocole OLSR que lorsqu'elle est intégrée dans une solution de routage centralisée et statique. Ceci peut s'expliquer par le fait que le protocole n'utilise que les MPRs pour relayer les paquets des flux, ce qui crée une topologie beaucoup moins dense que celle utilisée pour le routage statique. Par conséquent, les opportunités de trouver une route pouvant agréger les flux sont moindres, réduisant le gain potentiellement réalisable par rapport au routage sur le plus court chemin en nombre de sauts.

5.3 Simulations

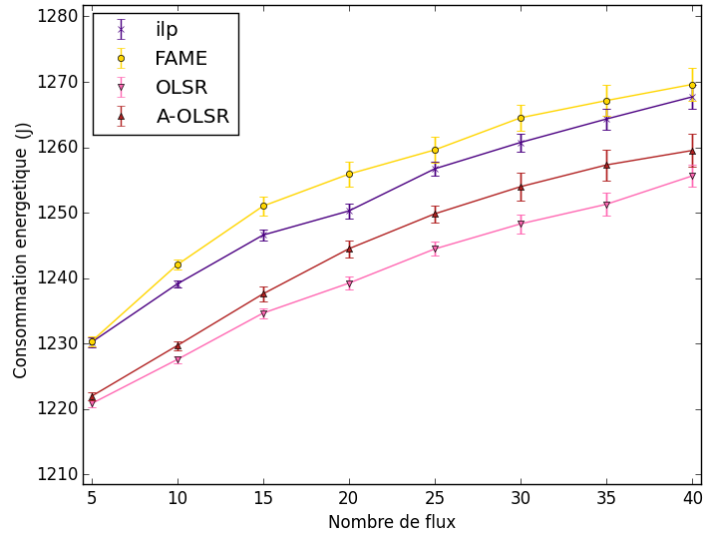


FIGURE 5.5 – Consommation énergétique globale du réseau en fonction du nombre de flux.

La Figure 5.5 montre la consommation énergétique en fonction du nombre de flux présents dans le réseau. Nous pouvons remarquer que la différence entre le routage sur le plus court chemin et le routage avec agrégation des flux est, en plus d’être extrêmement petite, à l’avantage du plus court chemin. La faible différence s’explique par le fait que nous n’avons pas éteint de nœuds dans ces simulations. Par conséquent, les nœuds non utilisés restent en mode Idle et continuent de consommer une quantité non négligeable d’énergie. L’augmentation de la consommation énergétique, elle, s’explique par le fait que pour agréger des flux, nous pouvons être amenés à attribuer à un flux un chemin plus long entre sa source et sa destination. Il s’agit ici du prix à payer pour pouvoir rerouter un flux vers des nœuds déjà utilisés. Ceci vient donc confirmer que l’agrégation seule n’est pas suffisante pour réduire la consommation énergétique. Elle doit être utilisée en complément d’une méthode d’extinction de nœuds pour y parvenir.

La Figure 5.6 montre l’énergie consommée par l’ensemble du réseau pendant la période de présence des flux. Nous pouvons constater que le fait d’éteindre les nœuds permet de réduire significativement la consommation énergétique. Comme vu précédemment, lorsque les nœuds non utilisés ne sont pas éteints, la consommation énergétique du réseau (OLSR et A-OLSR) évolue très lentement en fonction de la charge. Ceci confirme que la charge du réseau n’a que peu d’influence sur la consommation globale de ce dernier. En revanche, lorsque les nœuds non utilisés sont éteints, la réduction de la consommation énergétique est significative, allant jusqu’à plus de 50%.

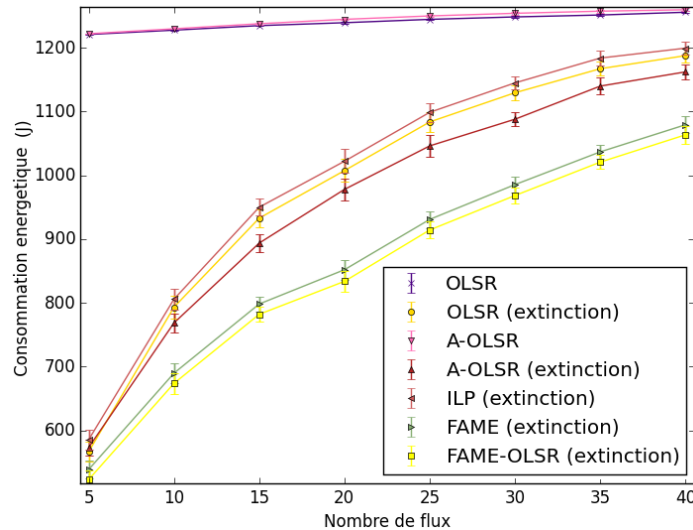


FIGURE 5.6 – Consommation énergétique globale en fonction du nombre de flux.

Le gain se réduit au fur et à mesure que le nombre de flux augmente puisque de plus en plus de nœuds doivent être utilisés pour router les flux, soit en tant que relais, soit en tant que sources ou destinations. Nous retrouvons, aussi, le faible gain observé avec OLSR lorsque l'agrégation est utilisée. Ceci vient confirmer que, même si notre métrique d'agrégation peut s'intégrer facilement au protocole OLSR et permettre une amélioration en matière de consommation énergétique, un effort supplémentaire d'intégration doit être fait en prenant davantage en compte les spécificités du protocole (en particulier le mécanisme de sélection des MPRs), afin d'obtenir une agrégation de flux plus efficace. Cette constatation est renforcée par les courbes relatives à ILP et FAME, montrant le gain pouvant être obtenu avec une solution de routage statique. Enfin, FAME-OLSR montre la consommation hypothétique du réseau dans le cas où les nœuds non utilisés n'auraient pas été calculés par le protocole OLSR mais grâce à la solution FAME. Cette courbe confirme, une nouvelle fois, que des mécanismes limitent la puissance du routage avec la métrique FAME dans le protocole OLSR.

La Figure 5.7 montre le taux de paquets non délivrés en fonction du nombre de flux présents dans le réseau. Nous pouvons remarquer ici que l'extinction des nœuds non utilisés n'a pas un grand impact sur le taux de paquets perdus. Ceci peut s'expliquer par le fait que l'extinction des nœuds a peu d'influence sur les routes empruntées par les flux, étant donné que les nœuds éteints sont calculés en fonction de l'algorithme de routage. En revanche, la différence se voit lorsque l'agrégation est utilisée ou non. En agrégeant les flux, le taux de perte de paquets augmente significativement. Ceci est conforme aux résultats obtenus dans le Chapitre 3, où il a été montré que l'agrégation de flux avait un impact négatif sur la capacité du réseau.

5.3 Simulations

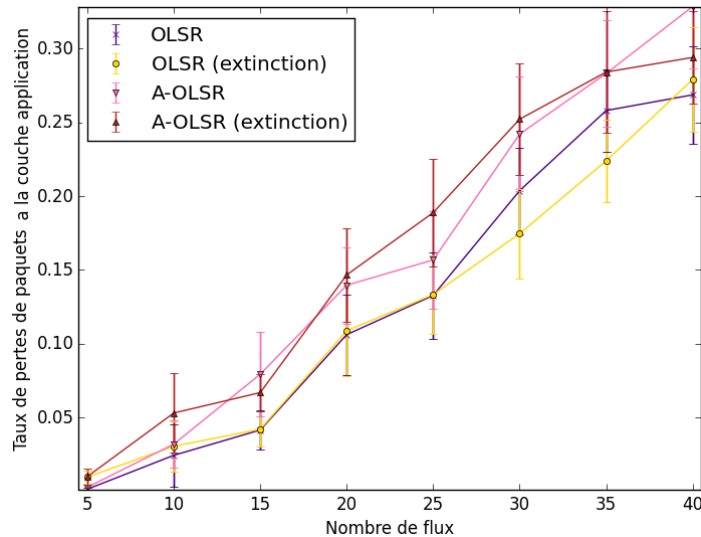


FIGURE 5.7 – Taux de paquets perdus au niveau application en fonction du nombre de flux.

Ceci peut s’expliquer par le cumul des pertes de paquets ayant lieu sur les chemins plus longs et par des problèmes de congestion dûs à l’augmentation de la charge locale des nœuds intermédiaires. Nous avons toutefois proposé dans les chapitres précédents des méthodes spécifiques afin de limiter ces effets négatifs.

Enfin, la Figure 5.8 montre le taux de paquets rejetés au niveau MAC. Il s’agit ici de paquets correctement reçus par la couche physique (puissance de signal suffisante), qui se sont trouvés rejetés par la couche MAC parce que des erreurs de transmission ont corrompu le paquet ou parce que ce dernier était destiné à un autre nœud. Ces résultats illustrent un certain type d’interférences que nous souhaitons pouvoir réduire afin d’améliorer la qualité des transmissions. Lorsque les nœuds non utilisés sont éteints, l’utilisation de l’agrégation des flux augmente le nombre de ces interférences. Ceci s’explique principalement par la longueur moyenne des flux qui entraîne une augmentation de la charge du réseau et du nombre de transmissions. Toutefois, nous pouvons remarquer que l’extinction des nœuds non utilisés permet une réduction des interférences. Et puisque l’agrégation de flux engendre une minimisation du nombre de nœuds utilisant le support de transmission, nous pouvons réduire significativement le nombre d’accès concurrents à ce dernier. Les nœuds restants sont donc moins soumis aux interférences des autres nœuds actifs du réseau. Toutefois, le trop faible écart entre OLSR et A-OLSR en termes d’agrégation ne permet pas d’obtenir une différenciation entre les deux solutions. En revanche, FAME-OLSR nous laisse entrevoir la possibilité d’obtenir une réduction très importante des interférences dans le cas d’une meilleure intégration de la métrique d’agrégation (protocole réactif, meilleure sélection des MPRs, etc.).

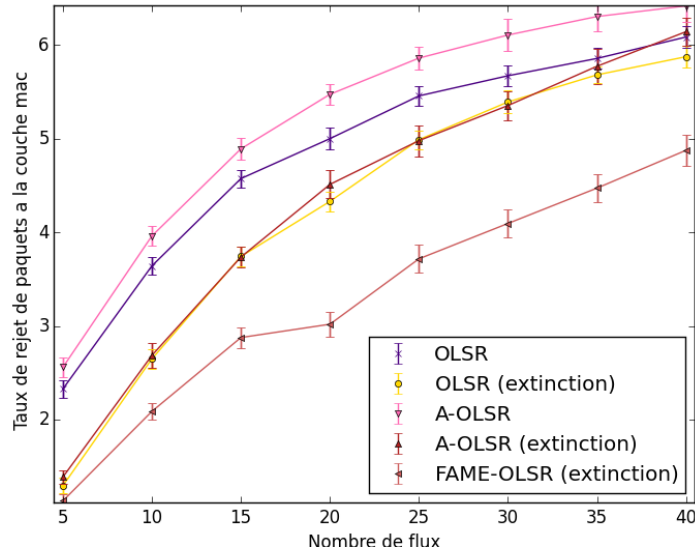


FIGURE 5.8 – Taux de paquets rejetés au niveau MAC en fonction du nombre de flux.

5.4 Conclusion

Notre métrique d'agrégation peut être intégrée dans des algorithmes et protocoles de routage existants, afin de réduire le nombre de nœuds utilisés dans le réseau et ainsi minimiser la consommation énergétique globale lorsque ces derniers sont éteints par une procédure d'extinction de nœud. Ce genre de procédure permet également de réduire l'exposition des nœuds aux interférences. Toutefois, ce chapitre a également mis en évidence certaines limites propres à l'agrégation de flux et à l'extinction des nœuds.

La concentration des flux est un problème déjà abordé dans le Chapitre 3. Pour rerouter des flux vers des nœuds déjà utilisés, il est nécessaire d'allonger la longueur des chemins empruntés par certains flux. Il en résulte une augmentation du nombre de transmissions globales dans le réseau, causant une augmentation de la charge totale. Le deuxième effet de l'agrégation de flux est le risque de voir apparaître de la congestion au niveau des nœuds restés allumés et chargés de retransmettre les flux. Les conséquences ont été montrés dans ce chapitre, notamment une augmentation du taux de perte des paquets de bout-en-bout.

Nous avons proposé une intégration de notre métrique d'agrégation dans le protocole de routage proactif OLSR et décrit la mise en œuvre. Les résultats montrent que le protocole OLSR, de par sa nature à réduire la connexité du réseau au moyen de son mécanisme de MPRs, ne permet pas de profiter pleinement de la puissance de notre métrique d'agrégation. Les résultats sont très encourageants et mettent en lu-

5.4 Conclusion

mière la nécessité d'étendre le mécanisme de sélection des MPRs pour tendre vers des performances quasi-optimales. À noter toutefois que nous avons étudié une intégration de FAME uniquement dans un protocole de routage proactif et qu'il serait intéressant d'envisager une intégration de notre métrique dans un protocole de routage réactif.

Chapitre 6

Conclusion et Perspectives

L'objectif de cette thèse consistait à proposer une méthode permettant de minimiser le nombre de nœuds impliqués dans le routage d'un ensemble de flux dans les réseaux sans fil multi-sauts. L'approche que nous avons proposée vise à offrir un effet de levier afin d'améliorer les solutions faisant de l'économie d'énergie en cherchant à éteindre ou à mettre en veille les nœuds du réseau. En effet, en minimisant le nombre de nœuds actifs, il devient possible d'éteindre davantage de nœuds et ainsi améliorer l'efficacité énergétique du réseau. L'utilisation des outils de la recherche opérationnelle a permis, dans un premier temps, de développer une solution optimale et centralisée à ce problème de routage. Cette solution permet, en théorie, de réduire la consommation énergétique globale du réseau de plus de 20% par rapport au cas où les flux sont routés en termes de plus court chemin en nombre de sauts (c'est-à-dire la solution adoptée la plupart du temps). Bien que cette première contribution ne soit pas adaptée pour une application concrète dans les réseaux sans fil multi-sauts, elle offre une borne supérieure optimale grâce à laquelle il devient possible d'évaluer d'autres solutions heuristiques, réalisables en pratique et moins complexes.

De ce travail est ressorti une seconde contribution, avec la définition d'une métrique de routage d'agrégation de flux, notée FAME. Cette métrique peut être intégrée dans des algorithmes de routage de type plus court chemin et permet de router des flux pour favoriser leur regroupement (agrégation) dans le voisinage de certains nœuds particuliers du réseau, appelés nœuds d'intérêt. Le développement de cette métrique a également été motivé par la volonté de rendre l'intégration de l'agrégation de flux dans les algorithmes et protocoles de routage existants la plus simple possible. Ceci a pu être vérifié par une expérimentation avec le protocole de routage proactif OLSR, laquelle a également mis en lumière certains défis liés à la fois à l'agrégation des flux et aux méthodes d'extinction des nœuds.

La principale limite liée à l'agrégation des flux est la concentration du trafic sur de petites zones du réseau. Dans les réseaux sans fil multi-sauts, ceci a pour effet de concentrer les interférences sur un plus petit nombre de nœuds et d'augmenter la charge du réseau sur l'ensemble du voisinage de ces derniers. Une version adaptative de FAME a ainsi été proposée. Basée sur une simplification d'un modèle d'interférence largement éprouvé par des travaux antérieurs, cette nouvelle métrique peut adapter l'agrégation des flux en fonction des niveaux de charges locales du réseau. Ainsi, si une zone du réseau voit son trafic trop augmenter, la métrique peut s'adapter de sorte à délester une partie de ce dernier sur une autre zone, moins chargée.

L'augmentation de la charge du réseau peut également être résolue par l'utilisation de codage réseau de type inter-flux, afin de réduire le nombre de transmissions dans le réseau. L'agrégation de flux crée des opportunités de codage permettant de bénéficier d'une bonne synergie entre les deux méthodes. L'étude menée sur ce point a mis en valeur une baisse significative de la surcharge engendrée par l'agrégation. De plus, la simplicité du modèle de codage réseau utilisé dans cette thèse permet d'envisager à la fois une implémentation concrète ou un gain potentiellement meilleur en considérant un modèle plus complexe. Enfin, le codage réseau intra-flux pourrait également présenter une synergie intéressante avec l'agrégation de flux afin d'améliorer la qualité des transmissions, voire d'envisager une utilisation combinée des deux principes. L'interaction entre le codage réseau et l'agrégation de flux reste donc un sujet ouvert sur bien des aspects et peut être l'objet de futur travaux.

Les expérimentations menées avec la métrique d'agrégation FAME ont montré que l'efficacité de cette dernière en termes de nombre de nœuds utilisés dans le réseau est grandement influencée par les nœuds qui sont sélectionnés pour agréger les flux. Ainsi, choisir ces nœuds en fonction de leur nature vis-à-vis des flux (source ou destination) permet d'obtenir une excellente agrégation des flux, proche de l'optimal. Ce choix, efficace lorsqu'il considère le trafic, pourrait offrir des résultats intéressants dans le cas où il prendrait davantage en compte la nature du réseau sur lequel FAME serait déployée. En effet, pour des réseaux mesh, une sélection statique des nœuds en fonction de leur rôle de point d'accès pour les utilisateurs et de passerelle Internet peut être envisagée. Les réseaux de capteurs pourraient éventuellement sélectionner les nœuds servant à la collecte des données ou disposant d'une configuration énergétique plus efficace. Ces aspects techniques propres à certains types de réseaux seraient donc également intéressants à étudier par la suite.

Par ailleurs, il est important de rappeler que l'agrégation de flux n'est pas directement responsable de la réduction de la consommation énergétique globale du réseau. Elle n'offre qu'un effet de levier visant à améliorer l'efficacité d'algorithmes cherchant à réduire la consommation énergétique en éteignant des nœuds non utilisés. Par conséquent, l'efficacité énergétique de l'agrégation de flux dans les algorithmes et protocoles de routage existants reste fortement corrélée à la nature de ces derniers et à leurs interactions avec les méthodes d'extinction des nœuds. Ainsi, les expérimentations faites sur l'intégration de la métrique FAME dans le protocole de routage OLSR ont montré des résultats qu'il est important de prendre en considération. Premièrement, l'efficacité de FAME n'est pas aussi importante que ce que montraient les études faites par simulation avec des algorithmes de routage statiques. Le protocole OLSR, bien que permettant de fournir avec un coût réduit les informations nécessaires au calcul de la métrique d'agrégation, limite les performances attendues par FAME. Deuxièmement, l'extinction des nœuds peut avoir des conséquences sur le protocole de routage. Par exemple, une méthode d'extinction de nœuds de type Duty Cycle peut être la cause de problèmes de convergences de l'algorithme de routage, provoqués par les nombreux changements générés sur la topologie du réseau. Une solution à ce problème pourrait bien se trouver dans les systèmes de type Wake up Radio qui permettraient aux nœuds du réseau de demander l'activation d'un ou de plusieurs voisin(s). Ceci permettrait de résoudre la question classique du temps de mise en veille qui se pose dans la recherche sur les méthodes Duty Cycle.

Enfin, puisque le travail accompli dans cette thèse a pu mettre en évidence l'applicabilité d'une solution quasi-optimale pouvant minimiser le nombre de nœuds utilisés dans un réseau sans fil multi-sauts, deux axes de recherche seraient intéressants à étudier à l'avenir. Le premier serait sur l'effet de la technique d'agrégation de flux, présentée dans cette thèse, sur les réseaux filaires. Ces derniers étant beaucoup moins contraints que les réseaux sans fil multi-sauts, tant en termes de capacité et d'interférence que de mobilité, d'autres applications et bénéfices pourraient être obtenus grâce à l'agrégation. Le second axe de recherche consisterait à réutiliser les résultats de ces travaux afin d'améliorer les méthodes existantes d'extinction de nœuds. En effet, en prenant en compte les propriétés et l'applicabilité de l'agrégation de flux, il pourrait être possible d'avancer sur la résolution du problème de la durée d'extinction des nœuds des approches Duty Cycle ou d'offrir de nouvelles incitations à l'adoption des systèmes Wake up Radio qui sont aujourd'hui principalement limités par le coût (matériel et algorithmique) que représente leur déploiement.

Publications

1. Alexandre Laubé, Steven Martin, Dominique Quadri, Khaldoun Al Agha. **Optimal Flow Aggregation for Global Energy Savings in Multi-hop Wireless Networks**. In ADHOC-NOW (International Conference on Ad-hoc, Mobile, and Wireless Networks), Lille, France, July 2016.
https://link.springer.com/chapter/10.1007/978-3-319-40509-4_9
2. Alexandre Laubé, Steven Martin, Dominique Quadri, Khaldoun Al Agha, Guy Pujolle. **FAME : A Flow Aggregation MEtric for Shortest Path Routing Algorithms in Multi-Hop Wireless Networks**. In WCNC (IEEE Wireless Communications and Networking Conference), San Francisco, CA, USA, March 2017.
<http://ieeexplore.ieee.org/document/7925496/>
3. Alexandre Laubé, Steven Martin, Dominique Quadri, Khaldoun Al Agha, Guy Pujolle. **Procédé pour la minimisation de la consommation d'énergie globale dans un réseau de communication sans fil**. Brevet (en cours de dépôt).
4. Alexandre Laubé, Steven Martin, Dominique Quadri, Khaldoun Al Agha, Guy Pujolle. **Procédé de routage décentralisé avec agrégation de flux dans les réseaux sans fil**. Brevet (en cours de dépôt).

Bibliographie

- [1] Laura Marie Feeney and Martin Nilsson. Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1548–1557. IEEE, 2001.
- [2] D. Halperin, A. Sheth, and D. Wetheral. Demystifying 802.11n power consumption. *HotPower'10 Proceedings of the 2010 international conference on Power aware computing and systems*, 2010.
- [3] Chung-Sheng Li and Wanjiun Liao. Software defined networks. *IEEE Communications Magazine*, 51(2) :113–113, 2013.
- [4] Zhijing Qin, Grit Denker, Carlo Giannelli, Paolo Bellavista, and Nalini Venkatasubramanian. A software defined networking architecture for the internet-of-things. In *Network Operations and Management Symposium (NOMS), 2014 IEEE*, pages 1–9. IEEE, 2014.
- [5] David Johnson, Yin-chun Hu, and David Maltz. The dynamic source routing protocol (dsr) for mobile ad hoc networks for ipv4. Technical report, 2007.
- [6] Charles Perkins, Elizabeth Belding-Royer, and Samir Das. Ad hoc on-demand distance vector (aodv) routing. Technical report, 2003.
- [7] Nishant Gupta and Samir R. Das. *Energy-Aware On-Demand Routing for Mobile Ad Hoc Networks*, pages 164–173. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002.
- [8] Radhika D Joshi and Priti P Rege. Energy aware routing in ad hoc networks. *6th WSEAS International Conference on CIRCUITS, SYSTEMS, ELECTRONICS, CONTROL & SIGNAL PROCESSING*, pages 469–475, 2007.
- [9] M. Veerayya, V. Sharma, and A. Karandikar. SQ-AODV : A novel energy-aware stability-based routing protocol for enhanced qos in wireless ad-hoc networks. *MILCOM 2008 - 2008 IEEE Military Communications Conference*, Nov 2008.

BIBLIOGRAPHIE

- [10] M Tamilarasi and TG Palanivelu. Integrated energy-aware mechanism for MANETs using on-demand routing. *International Journal of Computer and Information Engineering*, 2(3) :212–216, 2008.
- [11] P Srinivasan, P Kamalakkannan, and SP Shantharajah. Stability and energy aware multipath routing for mobile ad hoc networks. *International Journal of Computer Applications*, 74(16), 2013.
- [12] J-E Garcia, A Kallel, K Kyamakya, K Jobmann, J-C Cano, and P Manzoni. A novel DSR-based energy-efficient routing algorithm for mobile ad-hoc networks. In *Vehicular Technology Conference, 2003. VTC 2003-Fall. 2003 IEEE 58th*, volume 5, pages 2849–2854. IEEE, 2003.
- [13] Meng Li, Lin Zhang, Victor OK Li, Xiuming Shan, and Yong Ren. An energy-aware multipath routing protocol for mobile ad hoc networks. *ACM Sigcomm Asia*, 5 :10–12, 2005.
- [14] Longbi Lin, Ness B Shroff, and R Srikant. Asymptotically optimal energy-aware routing for multihop wireless networks with renewable energy sources. *IEEE/ACM Transactions on networking*, 15(5) :1021–1034, 2007.
- [15] Floriano De Rango, Paola Lonetti, and Salvatore Marano. MEA-DSR : a multipath energy-aware routing protocol for wireless ad hoc networks. In *Advances in Ad Hoc Networking*, pages 215–225. Springer, 2008.
- [16] Thomas Clausen and Philippe Jacquet. Optimized link state routing protocol (olsr). Technical report, 2003.
- [17] Hakim Badis and Khaldoun Al Agha. Qolsr, qos routing for ad hoc wireless networks using olsr. *European Transactions on Telecommunications*, 16(5) :427–442, 2005.
- [18] Floriano De Rango, Marco Fotino, and Salvatore Marano. EE-OLSR : energy efficient olsr routing protocol for mobile ad-hoc networks. In *Military Communications Conference, 2008. MILCOM 2008. IEEE*, pages 1–7. IEEE, 2008.
- [19] C. K. Toh. Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks. *IEEE Communications Magazine*, 39(6) :138–147, Jun 2001.
- [20] C-K Toh, Hiroshi Cobb, and David A Scott. Performance evaluation of battery-life-aware routing schemes for wireless ad hoc networks. In *Communications, 2001. ICC 2001. IEEE International Conference on*, volume 9, pages 2824–2829. IEEE, 2001.
- [21] Dongkyun Kim, JJ Garcia-Luna-Aceves, Katia Obraczka, Juan-Carlos Cano, and Pietro Manzoni. Power-aware routing based on the energy drain rate for mobile

- ad hoc networks. In *Computer Communications and Networks, 2002. Proceedings. Eleventh International Conference on*, pages 565–569. IEEE, 2002.
- [22] Thomas Kunz. Energy-efficient variations of olsr. In *Wireless Communications and Mobile Computing Conference, 2008. IWCMC'08. International*, pages 517–522. IEEE, 2008.
- [23] Zhihao Guo, Shahdi Malakooti, Shaya Sheikh, Camelia Al-Najjar, Matthew Lehman, and Behnam Malakooti. Energy aware proactive optimized link state routing in mobile ad-hoc networks. *Applied Mathematical Modelling*, 35(10) :4715–4729, 2011.
- [24] Thomas Clausen, Christopher Dearlove, Phillippe Jacquet, and Ulrich Herberg. The optimized link state routing protocol version 2. Technical report, 2014.
- [25] Ahmed Loutfi and Mohammed Elkoutbi. A new efficient and energy-aware clustering algorithm for the olsr protocol. *International Journal of Computer Science and Network Security (IJCSNS)*, 15(7) :85, 2015.
- [26] Volkan Rodoplu and Teresa H Meng. Minimum energy mobile wireless networks. *IEEE Journal on selected areas in communications*, 17(8) :1333–1344, 1999.
- [27] Swetha Narayanaswamy, Vikas Kawadia, Ramavarapu S Sreenivas, and PRe Kumar. Power control in ad-hoc networks : Theory, architecture, algorithm and implementation of the compow protocol. In *European wireless conference*, volume 2002, pages 156–162. Florence, Italy, 2002.
- [28] Benjie Chen, Kyle Jamieson, Hari Balakrishnan, and Robert Morris. Span : An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. *Wireless networks*, 8(5) :481–494, 2002.
- [29] Sudipto Guha and Samir Khuller. *Approximation algorithms for connected dominating sets*, pages 179–193. Springer Berlin Heidelberg, Berlin, Heidelberg, 1996.
- [30] Vikas Kawadia and PR Kumar. Power control and clustering in ad hoc networks. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 1, pages 459–469. IEEE, 2003.
- [31] Steven Martin, Khaldoun Al Agha, and Guy Pujolle. Traffic-based topology control algorithm for energy savings in multi-hop wireless networks. *annals of telecommunications-Annales des télécommunications*, 67(3-4) :181–189, 2012.
- [32] Wei Ye, John Heidemann, and Deborah Estrin. An energy-efficient mac protocol for wireless sensor networks. In *INFOCOM 2002. Twenty-first annual joint conference of the IEEE computer and communications societies. Proceedings. IEEE*, volume 3, pages 1567–1576. IEEE, 2002.

BIBLIOGRAPHIE

- [33] Michele Magno, S Marinkovic, Bruno Srbinovski, and Emanuel M Popovici. Wake-up radio receiver based power minimization techniques for wireless sensor networks : A review. *Microelectronics Journal*, 45(12) :1627–1633, 2014.
- [34] Ya Xu, John Heidemann, and Deborah Estrin. Geography-informed energy conservation for ad hoc routing. In *Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 70–84. ACM, 2001.
- [35] Curt Schurgers, Vlasios Tsiatsis, and Mani B Srivastava. Stem : Topology management for energy efficient sensor networks. In *Aerospace Conference Proceedings, 2002. IEEE*, volume 3, pages 3–3. IEEE, 2002.
- [36] Alberto Cerpa and Deborah Estrin. Ascent : Adaptive self-configuring sensor networks topologies. *IEEE transactions on mobile computing*, 3(3) :272–285, 2004.
- [37] Lin Gu and John A Stankovic. Radio-triggered wake-up for wireless sensor networks. *Real-Time Systems*, 29(2-3) :157–182, 2005.
- [38] Heikki Karvonen, Juha Petäjäjärvi, Jari Iinatti, Matti Hämäläinen, and Carlos Pomalaza-Ráez. A generic wake-up radio based mac protocol for energy efficient short range communication. In *Personal, Indoor, and Mobile Radio Communication (PIMRC), 2014 IEEE 25th Annual International Symposium on*, pages 2173–2177. IEEE, 2014.
- [39] Joaquim Oller, Ilker Demirkol, Jordi Casademont, Josep Paradells, Gerd Ulrich Gamm, and Leonhard Reindl. Performance evaluation and comparative analysis of subcarrier modulation wake-up radio systems for energy-efficient wireless sensor networks. *Sensors*, 14(1) :22–51, 2013.
- [40] Marcel C Castro, Peter Dely, Jonas Karlsson, and Andreas Kessler. Capacity increase for voice over ip traffic through packet aggregation in wireless multihop mesh networks. In *Future Generation Communication and Networking (FGCN 2007)*, volume 2, pages 350–355. IEEE, 2007.
- [41] L Krishnamachari, Deborah Estrin, and Stephen Wicker. The impact of data aggregation in wireless sensor networks. In *Distributed Computing Systems Workshops, 2002. Proceedings. 22nd International Conference on*, pages 575–578. IEEE, 2002.
- [42] Mehdi Malboubi, Liyuan Wang, Chen-Nee Chuah, and Puneet Sharma. Intelligent sdn based traffic (de) aggregation and measurement paradigm (istamp). In *INFOCOM, 2014 Proceedings IEEE*, pages 934–942. IEEE, 2014.
- [43] Antonio De La Oliva, Albert Banchs, and Pablo Serrano. Throughput and energy-aware routing for 802.11 based mesh networks. *Computer Communications*, 35(12) :1433–1446, 2012.

- [44] Azrina Abd Aziz, Y Ahmet Sekercioglu, Paul Fitzpatrick, and Milosh Ivanovich. A survey on distributed topology control techniques for extending the lifetime of battery powered wireless sensor networks. *IEEE communications surveys & tutorials*, 15(1) :121–144, 2013.
- [45] Bin Ni, Naveen Santhapuri, Zifei Zhong, and Srihari Nelakuditi. Routing with opportunistically coded exchanges in wireless mesh networks. In *Wireless Mesh Networks, 2006. WiMesh 2006. 2nd IEEE Workshop on*, pages 157–159. IEEE, 2006.
- [46] Youghourta Benfattoum, Steven Martin, and Khaldoun Al Agha. Irocx : interference-aware routing with opportunistically coded exchanges in wireless mesh networks. In *Wireless Communications and Networking Conference (WCNC), 2011 IEEE*, pages 1113–1118. IEEE, 2011.
- [47] Piyush Gupta and Panganmala R Kumar. The capacity of wireless networks. *IEEE Transactions on information theory*, 46(2) :388–404, 2000.
- [48] Kamal Jain, Jitendra Padhye, Venkata N Padmanabhan, and Lili Qiu. Impact of interference on multi-hop wireless network performance. *Wireless networks*, 11(4) :471–487, 2005.
- [49] Rajarshi Gupta, John Musacchio, and Jean Walrand. Sufficient rate constraints for qos flows in ad-hoc networks. *Ad Hoc Networks*, 5(4) :429–443, 2007.
- [50] Russell C Walker. *Introduction to mathematical programming*. Pearson College Division, 1999.
- [51] SM Sinha. *Mathematical Programming : Theory and Methods*. Elsevier, 2005.
- [52] M Pióro, A Jüttner, J Harmatos, A Szentesi, P Gajowniczek, and A Mysek. Topological design of telecommunication networks nodes and links localization under demand constraints. *Teletraffic Science and Engineering*, 4 :629–642, 2001.
- [53] Xuan-Nam Nguyen, Damien Saucez, Chadi Barakat, and Thierry Turletti. Optimizing rules placement in openflow networks : trading routing for better efficiency. In *Proceedings of the third workshop on Hot topics in software defined networking*, pages 127–132. ACM, 2014.
- [54] Hakim Ghazzai, Elias Yaacoub, Mohamed-Slim Alouini, Zaher Dawy, and Adnan Abu-Dayya. Optimized lte cell planning with varying spatial and temporal user densities. *IEEE Transactions on Vehicular Technology*, 65(3) :1575–1589, 2016.
- [55] IBM. Cplex optimizer. www.ibm.com/software/commerce/optimization/cplex-optimizer/.
- [56] M Garey and D.S. Johnson. *Computers and Intractability : A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, 1979.

BIBLIOGRAPHIE

- [57] Robert J Vanderbei. *Linear programming*. Springer, 2015.
- [58] David R Morrison, Sheldon H Jacobson, Jason J Sauppe, and Edward C Sewell. Branch-and-bound algorithms : A survey of recent advances in searching, branching, and pruning. *Discrete Optimization*, 19 :79–102, 2016.
- [59] Michel Gondran and Michel Minoux. *Graphs and algorithms*. Wiley, 1984.
- [60] Y. Benfattoum, S. Martin, I. Gawedzki, and K. AlAgha. I2ASWP : Routing considering intra-flow interference consideration in ad hoc network. *Research report no. 1539, CNRS-Univ. Paris Sud(LRI)*, December 2010.
- [61] Fred Glover. Improved linear integer programming formulations of nonlinear integer problems. *Management science vol.22, no.4*, December 1975.
- [62] NS-3 Consortium. Network simulator 3 (ns-3). <https://www.nsnam.org/>.
- [63] A. W. Tucker C. E. Miller and R. A. Zemlin. Integer programming formulation of traveling salesman problems. *JACM*, June 1960.
- [64] Ding-Zhu Du, Bing Lu, Hung Q Ngo, and Panos M Pardalos. Steiner tree problems. *Encyclopedia of optimization*, 5 :227–290, 2009.

Titre : Agrégation de trafic pour réduire la consommation énergétique globale dans les réseaux sans fil multi-sauts

Mots clefs : réseau sans fil multi-sauts, économie d'énergie, optimisation linéaire en nombres entiers, extinction de nœuds, métrique de routage, OLSR, agrégation de flux

Résumé : Les incitations économiques et environnementales poussent dorénavant à intégrer des considérations énergétiques dès les premiers stades de développement des réseaux. Dans les réseaux sans fil multi-sauts, l'approche la plus courante consiste à répartir le trafic sur l'ensemble des nœuds afin de réduire équitablement la consommation énergétique de chacun, avec pour objectif la maximisation de la durée de vie du réseau. Or cette approche n'est pas optimale vis-à-vis de la consommation globale du réseau, le niveau d'activité d'un nœud n'ayant souvent que peu d'influence sur sa propre consommation. Nous montrons que la meilleure approche consiste à éteindre un maximum de nœuds. Parmi les travaux réalisés, nous avons développé une solution s'appuyant sur de la programmation linéaire en nombres entiers et des simulations de graphes et de réseaux.

La solution proposée permet ainsi une agrégation de flux optimale, réduisant significativement le nombre de nœuds utilisés dans le réseau. Afin de la rendre applicable en pratique, une heuristique est détaillée permettant une mise en œuvre distribuée grâce à l'utilisation d'une métrique de routage. Ainsi, tout algorithme de plus court chemin peut nativement réaliser de l'agrégation de flux efficace. Il est important de souligner que la qualité de service en termes de débit est respectée. En effet, les interférences sont prises en compte et il est possible de profiter de la synergie offerte par l'utilisation conjointe de l'agrégation de flux et du codage réseau. Nos résultats améliorent significativement les méthodes d'économie d'énergie basées sur l'extinction des nœuds en leur permettant d'en éteindre davantage.

Title : Traffic aggregation for global energy savings in multi-hop wireless networks

Keywords : multi-hop wireless networks, energy savings, integer linear programming, node turned off, routing metric, OLSR, flow aggregation

Abstract : Economic and ecologic incentives are now leading people to design networks with energetic considerations at early stages of development. Most of the works for multi-hop wireless networks tend to spread the traffic uniformly over the network to reduce the energy consumption of each node individually. However, considering that the traffic of a node doesn't impact significantly its energy consumption, this approach is not optimal regarding the global energy consumption of the network. Finding a way to turnoff as much node as possible seems then to be a better way. This PhD

thesis focus on routing a set of flows over a multi-hop wireless network while minimizing the number of used nodes. This is done by using a distributed metric that allows the shortest path routing algorithms to perform flow aggregation. Using integer linear programming and simulations, we proposed an efficient solution to aggregate flows to significantly reduce the number of nodes used in the network. It allows, then, to improve algorithms that reduce the energy consumption of networks by increasing the number of nodes that can be turned off.