Conception et mise en oeuvre d'une architecture de communication pour mini-drones civils

Ons Bouachir

Université Fédérale
Toulouse Midi-Pyrénées

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)

Présentée et soutenue par :
Ons BOUACHIR

le mardi 02 Décembre 2014

Titre :

Conception et mise en œuvre d'une architecture de communication pour mini-drones civils

---

École doctorale et discipline ou spécialité :
EDSYS : Informatique 4200018

Unité de recherche :
Groupe RESCO, Laboratoire TELECOM, École Nationale de l'Aviation Civile (ENAC)

Directeur/trice(s) de Thèse :

Thierry GAYRAUD
Fabien GARCIA

Jury :

Congduc PHAM,	Université de Pau,	Rapporteur
Thomas NOEL,	Université de Strasbourg,	Rapporteur
Mounir FRIKHA,	École supérieure de communication de Tunis,	Examinateur
Yacine GHAMRI-DOUDANE,	Université de La Rochelle,	Examinateur
Bastien MANCINI,	Delair-Tech	Invité
Fabien GARCIA,	ENAC	Directeur de thèse
Thierry GAYRAUD,	Université Toulouse 3 Paul Sabatier,	Directeur de thèse

Conception et mise en œuvre d'une architecture de communication pour mini-drones civils

Ons BOUACHIR

Résumé

Les drones, engins volants sans pilotes généralement appelé UAV pour *Unmanned Aerial Vehicle*, ont longtemps été un outil militaire. Ces dernières années, l'avènement de micro-drones, plus petits, légers et moins coûteux que leurs homologues a étendu l'utilisation des UAV au domaine civil.

Afin d'améliorer les performances des drones sur de telles missions, des recherches sont menées afin de rendre les drones coopératifs. Une flotte de drones coopératifs serait alors capable de réaliser plus rapidement des missions plus complexes en partageant les différentes tâches entre les UAV. Ce genre d'opération nécessite un niveau élevé de coordination entre les drones rendu possible par un échange continu d'information entre eux et avec leur station de contrôle. Cet échange permet aux contrôleurs d'ajuster la mission selon les circonstances. La communication est donc vue comme un enjeu majeur dans l'évolution des opérations de ce genre de systèmes. Le réseau ad hoc est une solution prometteuse pour faire communiquer les drones entre eux et avec la station sol. Ses capacités d'auto-organisation lui permettent de fonctionner dans des situations difficiles comme des «canyons urbains» ou après les catastrophes naturelles.

Au cours d'une mission, une flotte de drones coopératifs, ayant chacun un rôle différent des autres, échange une variété de types de messages envoyés entre les UAV sur la liaison descendante (des drones vers la station sol) et sur la liaison montante. Ces messages ont des priorités et des besoins en termes de la qualité de service (QdS) qui diffèrent selon la tâche accordée à l'émetteur. Ces besoins peuvent varier selon le contenu du message et au cours du temps avec l'évolution et le changement de la tâche. Par conséquent, la QdS doit être gérée par un module conscient de ces évolutions et capable de s'adapter à la variation des besoins et de fournir à chaque trafic sa demande durant toute l'opération.

Cette thèse, financée par le projet européen D3CoS, propose un système nommé DAN «*DCoS (Distributed Cooperative Systems) Ad hoc Network*», offrant des garanties en termes de différenciation de services selon une variété de critères choisis par les applications comme la tâche allouée à l'émetteur

et le contenu du message. Ce système permet de proposer aux agents collaboratifs différents niveaux de QdS en fonction du type d'application utilisée qui peut être variable au cours du temps.

DAN est composé d'un ensemble de modules permettant à une application d'exprimer ces besoins en termes de QdS. Il permet, par la suite, de classifier les paquets émis selon la demande et le changement des besoins de l'application dans le but de garantir la QdS demandée et de respecter les priorités de chaque classe.

Le système DAN devant être utilisé en environnement réel, il a été évalué à l'aide de deux moyens : la simulation et l'expérimentation réelle. Les simulations ont été réalisées dans un environnement réaliste en utilisant un modèle de mobilité reproduisant le mouvement des drones Paparazzi et nommé PPRZM pour (*PaPaRazzi Mobility*). PPRZM a été implémenté et validé par une étude comparative avec le modèle de mobilité (*Random Way-Point*) et des traces de mouvements réels de drones Paparazzi. Par la suite, DAN a été évalué par des expérimentations réelles. Il a été implémenté sur Linux embarqué sur des cartes Raspberry Pi intégrées aux drones Paparazzi. Plusieurs tests ont été réalisés permettant de valider les résultats obtenus par simulations. L'intégration de ce de système aux drones Paparazzi permettra de faire évoluer les missions impliquant une flotte de drones puisque avant cette réalisation, des fonctionnalités comme l'évitement de collisions ou l'allocation dynamique de tâches impliquaient directement la station sol, comme point de relai ou de prise de décision. Le système proposé permet de faciliter les échanges dans ce genre de missions tout en garantissant la QdS pour chaque drone.

Abstract

The UAVs (Unmanned Aerial Vehicle) have long been a military tool. In recent years, the advent of mini-UAVs, smaller, lighter and less expensive than their counterpart, extended the use of UAVs in civil domain. They offer the solution to many dangerous operations such as search and rescue of survivors after natural disasters, cartography, communication relaying between several areas of intervention of firefighters. UAVs replace piloted airplanes and helicopters which may avoid possible human casualties. In order to enhance the performance of UAVs, Researchesis being carried out to make cooperative aircraft. A swarm of cooperative UAVs sharing different tasks between drones, will be able to perform complex operation faster. Such a mission requires a high level of coordination between UAVs made possible by a frequent exchange of information between them and their control station.

The ad hoc network is a promising solution to have the UAV communicate with each other (inside the swarm) and with the control station (outside the swam). It allows UAVs to perform in difficult situations such as mountains and after natural disasters. During a mission, a swarm of cooperative UAVs, having different role, exchanges a variety of message type between the aircraft and their control station. These messages may have different needs in term of quality of service (QoS) which varies with the task given to the transmitter. The needs of each UAV may vary depending on the content of the message and over time with the evolution of its task. Therefore, QoS must be managed by a module aware of these evolutions and able to provide each traffic requirements throughout the operation.

This research work, financed by the European project D3CoS [1], proposes a communication system, called DAN (DCoS (Distributed Cooperative System) Ad hoc Network), for a swarm of cooperative UAVs (Unmanned Aerial Vehicle). This system provides a service differentiation according to a variety of criteria chosen by applications such as the role or the task assigned to the node, the message, etc. DAN architecture is a combination of several QoS

1. http ://www.d3cos.eu/

mechanisms created to be used in real environment with Paparazzi UAVs. Therefore, it was evaluated using two evaluation means : simulation and real experiments. Simulations were realized in realistic environment using a specific mobility model, called PPRZM (PaPaRaZzi Mobility). PPRZM is a stochastic mobility model that imitates Paparazzi UAVs movement designed and implemented for this study. It was validated using a comparative study between PPRZM, the well known mobility model RWP (Random Way-Point) and real Paparazzi UAV movement traces. In addition, DAN was evaluated by real experiments. It was implemented on embedded Linux on Raspberry Pi cards in order to be integrated to Paparazzi UAVs. Several tests were performed to validate the operation of the new Paparazzi communication system and the simulation results.

This work creates a new communication system for Paparazzi system that can provide the suitable medium to operate UAVs over an ad hoc network and to respect the need of each agent in term of QoS.

Remerciements

Après trois ans de travaux de recherche dans le cadre de cette thèse, je tiens en ces quelques lignes à exprimer ma reconnaissance envers tous ceux qui de près ou de loin y contribué.

J'exprime en premier lieu ma gratitude à mes directeurs de thèse. *Fabien Garcia* qui fut pour moi un directeur de thèse très attentif. Sa compétence, et sa rigueur scientifique m'ont beaucoup appris. *Thierry Gayraud* pour ses directives et ses conseils avisés tout au long de cette thèse. Mes sincère remerciement vont aussi à mon encadrant *Nicolas Larrieu* pour son soutien, ses relectures, sa gentillesse et ses conseils.
Soyez assurés de ma profonde gratitude.

Je remercie les messieurs *Congduc Pham* et *Thomas Noel* pour avoir rapporté ce manuscrit, ainsi que les messieurs *Mounir Frikha*, *Yacine Ghamri-Doudane* et *Bastien Mancini* pour avoir accepté de juger mon travail.

Mes vifs remerciements vont aussi à tous ceux qui m'ont aidé à réaliser les expérimentations réelles spécialement les membres de l'équipe drones de l'ENAC.

Je remercie toutes les personnes formidables que j'ai rencontrées à l'ENAC. Je pense particulièrement à tous les membres de l'équipe ResCo et spécialement à *Alain Pirovano*, notre adorable responsable d'équipe, et les autres membres : *Antoine*, *Frédéric*, *Jean-Aimé*, *Mickael*, *Quentin*, *Slim* et *Stephano*. Je ne saurais terminer sans remercier tous les stagiaires de cette équipe que j'ai croisés durant ces dernières trois années, spécialement ceux que j'ai eu l'occasion de superviser : *Chifa* et *Alinoé* et ceux avec qui j'ai partagé mon bureau durant la rédaction de ce manuscrit : *Rita* et *Gilles*.

Mes sincères remerciements et ma gratitude vont à mes amis qui m'ont accompagné durant ces trois ans : *Amel* (ma chère Moula ;)), *Amira*, *Hajer*, *Rihab*, *Madiha*, *Hasna* et *Anh*.

Enfin, les mots les plus simples étant les plus forts, j'adresse toute mon affection à ma famille : mes chers parents et mon frère *Hedi*. Je n'aurais rien fait de tout cela sans vous.

Table des matières

1 Introduction **1**
1.1 Contexte . . . 2
1.1.1 D3CoS . . . 2
1.1.2 Drones dans des opérations civiles . . . 4
1.1.3 Système Paparazzi . . . 8
1.2 Solutions pour les réseaux de communications . . . 10
1.2.1 Réseaux satellite . . . 10
1.2.2 Réseaux cellulaires . . . 11
1.2.3 Réseaux Ad hoc . . . 12
1.2.4 Discussion . . . 13
1.3 Contributions . . . 14
1.4 Structure du mémoire . . . 16

2 Réseaux ad hoc de drones **19**
2.1 Taxonomie des MANET . . . 20
2.1.1 Les VANET . . . 20
2.1.2 Les AANET . . . 21
2.1.3 Les réseaux Ad hoc de drones . . . 21
2.2 Mécanismes pour les réseaux ad hoc de drones . . . 25
2.2.1 Niveau MAC . . . 25
2.2.2 Niveau routage . . . 30
2.2.3 Mécanismes de gestion de la QdS . . . 39
2.3 Évaluation des réseaux ad hoc de drones . . . 51
2.3.1 Simulation . . . 51
2.3.2 Expérimentation réelle . . . 59

3 Architecture DAN **63**
3.1 Recueil d'exigences . . . 64
3.2 Classes de trafic . . . 67
3.3 L'architecture de DAN . . . 68
3.3.1 API DAN . . . 69

3.3.2 L'agent DAN 70
3.3.3 Le contrôleur d'admission 71
3.3.4 Le classificateur 71
3.4 La signalisation de DAN 72
3.5 Le Fonctionnement de DAN 73
3.5.1 API DAN 73
3.5.2 L'agent DAN 76
3.5.3 Le contrôleur d'admission 77
3.5.4 Le classificateur 78

4 Évaluation des performances de DAN par simulation 81
4.1 Modèle de mobilité PPRZM 82
4.1.1 Les mouvements des drones Paparazzi 82
4.1.2 PPRZM 85
4.1.3 Validation du modèle PPRZM 88
4.2 Évaluation de DAN 92
4.2.1 Évaluation du fonctionnement du Contrôleur d'admission 93
4.2.2 Étude des minuteurs 97
4.2.3 Performances générales de DAN 106
4.2.4 Évaluation de l'impact de variation de la charge du réseau 109
4.2.5 Évaluation avec d' autres protocoles de routage 112

5 Implémentation et plate-formes de tests 117
5.1 Outils utilisés 118
5.2 Implémentation de DAN 119
5.3 Résultats et discussion 121
5.3.1 Validation du fonctionnement du nouveau système de communication Paparazzi 123
5.3.2 Évaluation de DAN 127

6 Conclusions et perspectives 143
6.1 Travaux réalisés 143
6.1.1 L'architecture DAN 144
6.1.2 Intégration DAN au système Paparazzi 145
6.2 Perspectives 147
6.2.1 Perspectives réseaux 147
6.2.2 Opérations pour les drones 149

A Annexe : Outils de simulation 167

Table des figures

1.1 Communication coopérative à travers un drone relais 6
1.2 Retransmission de communications par des drones collaboratifs 7
1.3 Communications dans un système de drones coopératifs 8
1.4 Communications satellites 10
1.5 Communications cellulaires 12
1.6 Réseau Ad hoc 13

2.1 Réseau Ad hoc véhiculaire 20
2.2 Communications ad hoc et communications centralisées 26
2.3 Le mécanisme CSMA/CA 27
2.4 Le scénario de LODMAC 29
2.5 Les MPRs 33
2.6 La transmission géographique *gloutonne* 34
2.7 Mécanisme du protocole RGR 37
2.8 Méthode de groupge 38
2.9 Architecture hybride utilisée avec le protocole EHSR 39
2.10 Mouvements de RWP 53
2.11 Mouvements de RD 54
2.12 Mouvements de Gauss Markov avec trois valeurs de α 55
2.13 Le principe de SRCM 56

3.1 Architecture de DAN 70
3.2 Format des messages *Reserve* et *Error* 72
3.3 Format du message *Close* 73
3.4 Exemple de fonctionnement de l'API DAN 75
3.5 Diagramme de séquencement de l'établissement et le rafraichissement des réservations 77
3.6 Traitement de paquets par DAN 79

4.1 Le mouvement *rectiligne* 82
4.2 Le mouvement *circulaire* 83

4.3 Le mouvement *oblong* 84
4.4 Le mouvement *Huit* 84
4.5 Le mouvement *balayage* 85
4.6 Traces réelles de trois drones Paparazzi 85
4.7 Altitudes des trois drones Paparazzi utilisés dans le graphe 4.6 86
4.8 Fonctionnement de PPRZM 87
4.9 Exemple de calcul de métrique *Frequency* 89
4.10 Trajectoire suivie selon PPRZM 91
4.11 Comparaison géométrique 93
4.12 Comparaison des performances du réseau 94
4.13 Débits mesurés durant la simulation pour chaque flux de trafic 97
4.14 Taux de perte mesurés avec chaque valeur de WTC pour les trois classes de trafic 99
4.15 Délais de bout en bout mesurés avec les différentes valeurs de WTC 100
4.16 Débits de transmission mesurés avec la variation de la durée WTC 101
4.17 Rapports entre le nombre de paquets de signalisation envoyés et le nombre total de paquets Premium émis mesurés avec les différentes durées de WTC 102
4.18 Taux de perte mesurés pour chaque classe de trafic durant les différents tests 103
4.19 Délais mesurés pour chaque classe de trafic durant les différentes simulations 104
4.20 Débit de transmission mesurés pour les différentes classes de trafic 104
4.21 Nombre de messages de signalisation échangés au cours de chaque simulation 105
4.22 Nombre de paquets par chaque file d'attente 107
4.23 Débit mesuré durant la simulation pour les deux classes Premium et Best-effort 108
4.24 Taux de perte mesuré pour chaque classe de trafic 110
4.25 Délai mesuré pour chaque classe de trafic 111
4.26 Moyennes de débit de transmission mesurées pour les deux classes de trafic Premium et Best-effort 112
4.27 Taux de perte mesurés pour les différentes classes de trafic . . 113
4.28 Délai mesuré pour chaque classe de trafic 113
4.29 Débits mesurés pour les deux classes Best-effort et Premium avec les protocoles de routage OLSR et DSDV 114

5.1 Quadricoptère Paparazzi 118

5.2 Carte Raspberry Pi 118
5.3 Une clé Wi-Pi 119
5.4 l'intégration de DAN au système Paparazzi 120
5.5 Architecture du système Linux avec DAN 121
5.6 Extrait du diagramme de classe de l'agent DAN 122
5.7 Trajectoire des drones au cours de l'expérimentation 123
5.8 Capture d'écran de la station de contrôle 124
5.9 Les valeurs TTL des paquets des deux drones G1 et HEN1 . . 125
5.10 Débit du trafic descendant 126
5.11 Scénario des expérimentations 128
5.12 Débits mesurés pour les différents flux de trafic 130
5.13 Taux de perte mesurés pour chaque classe de trafic avec et sans DAN 132
5.14 Délais mesurés pour chaque classe de trafic avec et sans DAN 133
5.15 Débit mesuré durant l'expérimentation sans DAN pour les classes Premium et Best-effort 134
5.16 Débit mesuré durant l'expérimentation avec DAN pour les classes Premium et Best-effort 135
5.17 Débits mesurés pour les classes Premium et Best-effort 137
5.18 Débit mesuré durant une expérimentation 138
5.19 Nombre de paquets ICMP échangés 139
5.20 Débits mesurés pour les deux classes Premium et Best-effort . 140

A.1 Captures écran du simulateur OMNET++ 168
A.2 Architecture d'un nœud utilisant le système DAN 169
A.3 Architecture du module DAN_classificateur 169

Liste des tableaux

1.1 Avantages ou inconvénients des systèmes de communication . 14

2.1 Comparaison entre réseau de capteurs aériens et un réseau ad hoc de drones pour les systèmes multi-drones 23
2.2 Comparaison VANET et UAANET 24
2.3 Les protocoles de routage pour les UAANET 40

4.1 Scénario de la simulation 92
4.2 Premier scénario 95
4.3 Taux de perte mesuré pour chaque Flux (%) 95
4.4 Délai de bout en bout mesuré pour chaque flux (ms) 96
4.5 Temps d'attente dans la file d'attente pour chaque flux (ms) . 96
4.6 Débits mesurés pour chaque flux de trafic 96
4.7 Deuxième scénario 98
4.8 Troisième scénario 106
4.9 Taux de perte mesuré pour chaque classe de trafic (%) 106
4.10 Délai de bout en bout mesuré pour chaque classe de trafic . . 108
4.11 Débit mesuré pour les deux classes de trafic Premium et Best-effort (Kbits/s) 108
4.12 Quatrième scénario 109

5.1 Débit généré par les drones en kbits/sec 126
5.2 Taux de perte (%) mesurés durant les différentes périodes de l'expérimentation 127
5.3 Taux de perte mesuré pour les différents flux de trafic (%) . . 129
5.4 Délais de bout en bout mesurés pour les flux envoyés (ms) . . 130
5.5 Taux de perte dues à la congestion du réseau 132
5.6 Débit mesuré pour les classes Premium et Best-effort durant les deux expérimentations (kbits/s) 135
5.7 Taux de perte mesuré pour chaque classe de trafic (%) 136
5.8 Délai mesuré pour chaque classe de trafic 137

5.9 Débits mesurés pour les deux classes Premium et Best-effort (kbits/s) 137
5.10 Délai mesuré pour chaque classe de trafic (ms) 140

Chapitre 1

Introduction

Contents

1.1 Contexte 2
1.1.1 D3CoS 2
1.1.2 Drones dans des opérations civiles 4
1.1.3 Système Paparazzi 8
1.2 Solutions pour les réseaux de communications . 10
1.2.1 Réseaux satellite 10
1.2.2 Réseaux cellulaires 11
1.2.3 Réseaux Ad hoc 12
1.2.4 Discussion 13
1.3 Contributions 14
1.4 Structure du mémoire 16

Introduction

Dans le cadre du projet européen, D3CoS [1], ce travail de thèse étudie l'architecture de communication appropriée pour une flotte de drones. Ce genre de systèmes présente des améliorations pour les performances des opérations ordinaires et classiques de drones puisque plusieurs appareils collaborent entre eux afin de faciliter leur mission commune. Cette collaboration nécessite un niveau élevé de coordination entre les différents aéronefs rendu possible par un échange continu e différents types de données. Par conséquent, un système de communication fiable doit être utilisé pour faciliter la tâche des

1. Designing Dynamic Distributed Cooperative Human-Machine Systems : http ://www.d3cos.eu/

drones. La proposition de cette thèse est étudiée par simulation ainsi que par des expérimentations réelles sur des mini-drones *Paparazzi* présentés dans ce chapitre.

Ce chapitre présente en premier lieu le contexte de cette thèse. D'abord, il introduit le projet D3CoS et ses objectifs. Puis il présente le domaine civil d'application de flottes de drones et donne des exemples de missions coopératives. Ensuite, il introduit le système d'auto-pilote Paparazzi.
Dans la deuxième partie de ce chapitre, différents systèmes de communication sont présentés afin de trouver le système le mieux adapté aux missions coopératives d'une flotte de drones. Par la suite, la contribution de cette thèse est introduite dans la troisième sous-section. Finalement, la quatrième sous-section présente la structure de ce mémoire.

1.1 Contexte

1.1.1 D3CoS

Dans les dernière décennies, la demande sur les transports (terrestres, aériens et maritimes) a connu une augmentation remarquable et rapide. Aujourd'hui, ils sont devenus un facteur clef de la société modernes.

La relation classique entre l'humain et la machine est basée sur deux facettes du transport : le contrôle des véhicules et le contrôle du trafic. Les conducteurs des automobiles, les pilotes des aéronefs, les contrôleurs de la circulation aérienne et les officiers nautiques sont les responsables de la circulation des véhicules en toute sécurité même dans les zones les plus congestionnées ce qui peut causer, dans certaines situations, des dégâts humains et matériels à causent des capacités cognitives limitées de l'être humain.

Aujourd'hui des innovations technologiques ont permis l'introduction de systèmes d'assistance automatique dans ces tâches, apportant une interaction complexe entre humains et machines.

Les mauvaises utilisations et les défauts de maitrise de ces innovations peuvent être aussi la cause des nouvelles erreurs humaines sources d'incidents voir d'accidents et des conséquences désastreuses par exemple : la collision aérienne d'Überlingen [2]. La seule augmentation de l'automatisation des tâches n'est pas suffisante pour résoudre ces problèmes. Ainsi, un meilleur niveau de coopération entre humains et machines avec un partage d'autorité entre eux doit aussi être envisagé.

Il manque aujourd'hui les éléments clefs pour de nouvelles méthodes, des techniques et des outils qui vont au-delà des systèmes d'assistance tradition-

2. http ://www.securiteaerienne.com/la-collision-dueberlingen/

nels et répondent à l'ensemble des processus de développement des systèmes coopératifs. Ces éléments clefs doivent être basés sur une perspective multi-agents afin de relever les défis posés par les futurs environnements de circulation coopérative.

Le projet européen D3CoS, financé par l'initiative commune ARTEMIS [3], vise à développer des méthodes, des techniques et des outils (MTT : Methods, Technics and Tools) permettant de concevoir, développer et évaluer des systèmes coopératifs multi-agents offrants aux opérateurs humains et aux machines les moyens de partager les tâches assignées au système dans son ensemble. D3CoS vise à intégrer ces MTTs dans les processus de développement du système industriel pour soutenir le développement de systèmes abordables et innovants de coopération homme-machine.

Ces systèmes coopératifs dynamiques distribués homme-machine sont appelés DCoS (Distributed Cooperative Humain-Machine System).

Pour faire face à ce changement de paradigme d'une manière réaliste, D3CoS s'est limité aux systèmes coopératifs dans le domaine des transports dont deux tâches globales sont affectées :

- **Contrôle d'un seul véhicule** : Perspective de bord, où un véhicule effectue les tâches attribuées par un agent de bord(i.e. Un pilote de ligne avec un avion).
- **Contrôle d'un trafic de véhicules** : Perspective de la circulation, où un ensemble de véhicules est en compétition pour l'espace et le temps et ils résolvent leurs conflits en effectuant une collaboration spatio-temporelles (i.e. Un pilote contrôlant une flotte de drones).

D3CoS cible quatre domaines d'application : les avions pilotés, les véhicules aériens sans pilote (drones), l'automobile et le maritime.

Dans notre étude, nous allons nous focaliser sur le domaine des drones puisque depuis quelques années, l'avènement de mini-drones, plus petits et moins couteux que leurs homologues, a étendue l'utilisation de ces aéronefs au domaine civil afin de faciliter et améliorer les performances de certaines missions critiques.

Les missions civiles de drones sont présentées dans la section suivante.

3. Advanced Research and Technology in EMbedded Innovation Systems http ://www.artemis-ju.eu/

1.1.2 Drones dans des opérations civiles

Les drones, les engins volants sans pilotes (généralement appelés UAV pour *Unmanned Aerial Vehicle*), sont utilisés pour la première fois vers la fin de la première guerre mondiale par l'armée américaine. L'idée était de guider des avions pilotés à distance sans le pilote dans le cockpit.

Depuis, les drones sont devenus un sujet d'innovation dans différents domaines et pour plusieurs types de missions [Blo]. Ils ont longtemps été un outil militaire, cependant, ces dernières années, leur utilisation a été étendue au domaine civil. Ils présentent une solution prometteuse pour les missions les plus dangereuses, délicates et inadaptées aux pilotes humains. Ce qui permet de sauver des vies humaines puisqu'ils permettent de remplacer des avions et des hélicoptères pilotés.

Grâce au progrès technologiques, les drones, aujourd'hui, existent en plusieurs dimension [dFB08] pour pouvoir s'adapter aux différents types de mission. Il en existe des aéronefs petits, légers, moins coûteux et capables de remplacer les humains dans plusieurs missions civiles comme la recherche et le sauvetage des survivants après les catastrophes. En outre, ils sont utilisés dans plusieurs opérations de contrôle de l'environnement et des phénomènes naturels comme les émissions de gaz ou de liquide.

Depuis quelques années, les chercheurs conçoivent des applications pour les drones pour pouvoir opérer d'une manière plus autonome afin d'améliorer les performances de ces systèmes et de simplifier au maximum la tâche de l'opérateur humain.

1.1.2.1 Opérations traditionnelles

Grâce aux capteurs qui peuvent êtres intégrés dans l'aéronef, les drones sont utilisés dans différentes missions. La première utilisation d'un drone dans une mission civile était en 1969 par la NASA. Depuis, ils sont engagés dans plusieurs tâches gouvernementales dont par exemple les opérations de recherche des survivants après les catastrophes naturelles, les accidents, notamment dans les conditions les plus difficiles comme les conditions métrologiques et les localisations géographiques inaccessibles ou dangereuses [LRGM10]. En outre, ils sont utilisés dans des opérations policières comme le contrôle de la circulation routière. En effet, en Novembre 2010, la police allemande a utilisé pour la première fois un drone pour contrôler des manifestations civiles [4].

Ces aéronefs sont utiles aussi pour les pompiers. Grâce aux capteurs infrarouges qui peuvent détecter le feu, les drones sont utilisés pour découvrir les flammes dans les forêts dans le but d'informer les pompiers à temps. De

4. http ://www.bigbrotherawards.de/2011-en/.pol

plus, ils servent dans plusieurs autres tâches comme la collecte des informations utilisées pour l'évaluation des dommages ou pour la création et le suivi de la cartographie du feu en temps-réel ou comme la rediffusion des communications entre les pompiers et le centre de secours [CBM+05]. Les autorités américaines ont déjà utilisé le drone *Predator-B* modifié avec une camera infrarouge afin de pouvoir aider les pompiers à gérer le feu des forêts en Californie en 2007 [5]. L'aéronef transmettait des données précises en temps réel permettant d'anticiper la propagation du feu et de mieux exploiter les ressources.

Les drones sont utilisés aussi dans des missions de surveillance de grandes zones agricoles et même pour la pulvérisation des récoltes [6]. Ils peuvent servir, aussi, dans des tâches commerciales comme l'acquisition de données et d'images d'endroits inaccessibles et/ou dangereux, comme les régions sinistrées après les catastrophes naturelles. Les images prises par le drones peuvent servir pour la création de cartes géographiques et même pour la réalisation de films [7]. Les drones marquent leur présence aussi dans plusieurs domaines de recherches scientifiques comme l'environnement [8], l'atmosphère [9], la pollution [CSGM13, XLH+13], etc. Ils sont utilisés pour réaliser des expérimentations et prendre des mesures comme la température et l'humidité.

En outre, ils peuvent servir dans le domaine des communications et aider à retransmettre les échanges entre deux groupes d'utilisateurs éloignés comme en figure 1.1. En effet, à cause de leur portée limitée, les réseaux sans fil peuvent avoir des problèmes de connexion quand un équipement ou un groupe d'équipements tentent d'envoyer une information vers d'autres utilisateurs éloignés, en dehors de la portée de transmission de l'émetteur. Ainsi, le drone améliore la connectivité dans le réseau en re-émettant les messages transmis entre les deux extrémités.

Pour ce genre de missions, les drones doivent être contrôlés par un agent central, qui permet de diriger les opérations en contrôlant son évolution et l'état de l'aéronef. Cet agent pilote et donne les instructions nécessaires aux aéronefs en passant par une seule liaison de communication entre chaque drone et la station de contrôle. Ces échanges sont assurées, par des communications point à point et point à multi-point (en étoile).

5. http ://machinedesign.com/news/newest-forest-firefighter-predator-uav
6. http ://www.gizmag.com/uav-crop-dusting/27974/ //2012
7. http ://www.fastcompany.com/1816578/unmanned-drones-go-from-afghanistan-to-hollywood //2012
8. http ://www.aero-news.net/index.cfm ?do=main.textpost&id=aa628485-ba35-4bfe-86ba-c94ea52ac38b //2005
9. http ://press.scanex.ru/index.php/en/news/item/4008-atmo //July 2013

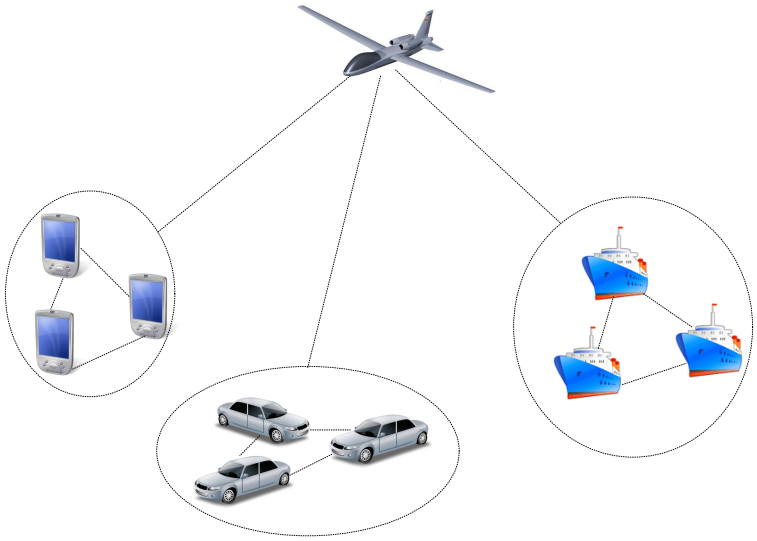


FIGURE 1.1 – Communication coopérative à travers un drone relais

1.1.2.2 Opérations coopératives

Afin d'améliorer les performances des drones sur ces missions, des recherches sont menées afin de rendre les drones coopératifs. Une flotte de drones coopératifs serait alors capable de réaliser plus rapidement des missions plus complexe par le partage de différentes tâche entre eux. Les drones peuvent être responsables de l'évolution de l'opération par l'acquisition de nouveaux comportement intelligents comme l'évitement de collisions et la formation autonome du vol [RZH+04].

La coopération entre les drones permet de faciliter les missions usuelles et d'améliorer ses performances [YCBE10]. Ayant différentes tâches, ils se complètent en collaborant pour atteindre un but commun comme par exemple les missions de retransmission des communications (figure 1.2), de création de cartes géographiques, de surveillance et les opérations de recherche et de sauvetage.

Cette collaboration est rendue possible par un niveau élevé de coordination entre les drones atteint par un échange continu de données entre les drones tout au long de la mission. Par exemple, dans [RZH+04], une mission de surveillance d'une large zone géographique est présentée pour être réalisée par un groupe de drones coopératifs. En échangeant des informations sur leurs positions géographiques, chaque aéronef peut calculer son itinéraire en fonction de celui des autres. Cet algorithme présente une meilleure performance en termes de rapidité et fiabilité par rapport à l'opération classique réalisée par un seul drone.

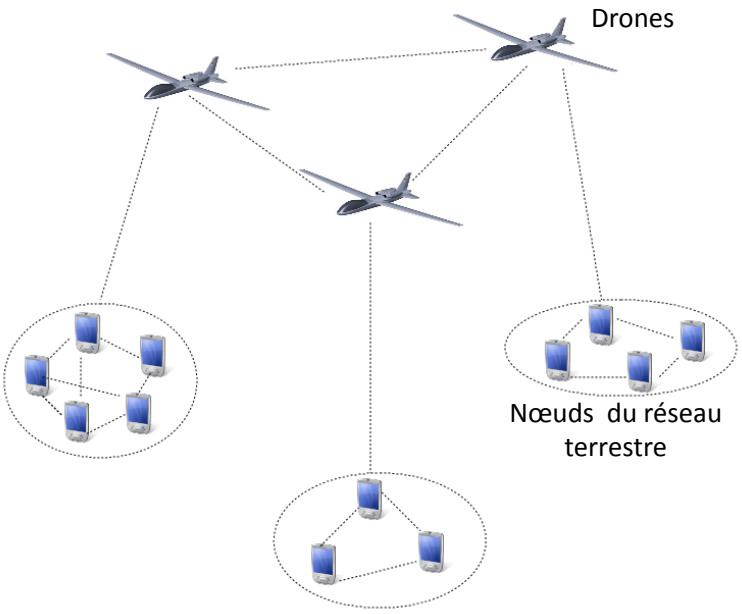


FIGURE 1.2 – Retransmission de communications par des drones collaboratifs

Plusieurs autres propositions ont été étudiées afin d'avoir une flotte de drones plus autonome qui peut se partager l'autorité et prendre des décisions quand c'est nécessaire et même d'allouer les différentes tâches entre les aéronefs. En effet, dans [MYM01], les auteurs ont présenté une méthodologie pour créer une stratégie de recherche coopérative mise en place par un groupe de drones. Cette méthodologie s'appuie sur l'échange des informations concernant la mobilité de chaque aéronef. Ces informations permettent aux drones d'avoir une idée de ce qui se passe autour d'eux et de calculer et de mettre à jour leurs chemins de recherche.

De plus, un algorithme d'estimation et de localisation des cibles pour un groupe de drones collaboratifs est proposé dans [BVH07]. Cette méthode présente de meilleurs résultats que ceux qui pourraient être obtenus avec un seul drone.

Le projet CARUS [CLM+11] étudiait aussi l'utilisation d'une flotte de drones. Son but était de montrer la faisabilité de l'utilisation de drones collaboratifs communicants dans une mission de surveillance de quelques points d'incident dans le but de résoudre des problèmes techniques et humains. L'échange des messages, dans cette étude, est réalisé par une diffusion asynchrone entre les drones.

La communication entre les agents joue un rôle très important dans ce

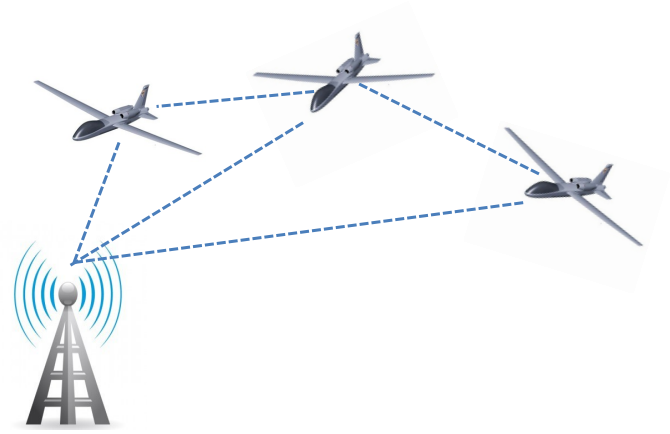


FIGURE 1.3 – Communications dans un système de drones coopératifs

genre d'opération puisque l'évolution de la mission peut être affectée ou décidée par l'évolution d'une seule tâche.

Dans ce genre de système, deux types de liens de communications sont nécessaires : le lien entre les drones (drone-drones) et la liaison de communication entre les drones et la station de contrôle (drone-sol) comme illustré par la figure 1.3.

Cette nouvelle tendance d'utilisation des drones attirent plusieurs experts et amateurs de ces engins. Ils essaient, d'intégrer ces fonctionnalités à leurs systèmes, et de créer de nouvelles applications et de nouvelles missions. Le système Paparazzi, est l'un de ces systèmes qui ne cesse d'évoluer pour intégrer des fonctions facilitant l'application des opérations coopératives.

1.1.3 Système Paparazzi

Créé par l'équipe drones de l'ENAC [10] en 2003, Paparazzi [Pap] est un système complet de logiciels et de matériel libre (*open source*) permettant de gérer des systèmes aériens sans pilotes (UAS pour *Unmanned Aerial System*). Il regroupe un auto-pilote pour diriger des aéronefs à voilure fixe ainsi que des *multicopters*, et une station sol qui contient des logiciels de planification de mission et de surveillance en utilisant une liaison de données bidirectionnelle pour le contrôle et la télémétrie.

10. École Nationale de l'Aviation Civile

Aujourd'hui Paparazzi est utilisé par des universités, des industriels ainsi que des amateurs partout dans le monde dans une grande variété d'aéronefs, dont plusieurs ont été conçus spécifiquement autour du système Paparazzi.

Paparazzi permet de gérer et de contrôler simultanément un ensemble de drones et continue à évoluer pour faciliter la tâche de contrôle de plusieurs aéronefs par un seul opérateur. De plus, il permet de développer des applications civiles pour les drones grâce aux différents types de capteurs comme par exemple les appareils photo, les caméras vidéo (en rajoutant un émetteur de vidéo) et les capteurs météorologique.

Afin de communiquer avec l'aéronef, l'auto-pilote Paparazzi utilise une liaison bidirectionnelle pour garantir une télémétrie en temps-réel et pour pouvoir faire des réglages et envoyer des commandes durant le vol.

La liaison descendante (de l'aéronef vers la station sol), est consacrée à la télémétrie. Elle contient des informations sur l'état de l'engin pour permettre l'évolution de la mission de contrôle comme par exemple la position géographique GPS de l'avion qui est envoyée et visualisée en temps réel au niveau de la station sol.

Sur la liaison montante (de la station sol vers l'aéronef), les messages de contrôle à distance ou les commandes sont envoyés afin d'interagir avec le drone : par exemple, le changement de la direction ou du point d'acheminement de l'aéronef.

Le dispositif de liaison de données le plus utilisé pour les systèmes Paparazzi est la série XBee. XBee s'appuie sur le standard 802.15.4 [Soc03], la base du Zigbee. Il fournit des communications point-à-point et point-à-multipoint (en étoile) avec un débit qui ne dépasse pas les 250 kbits/s. Ainsi, pour émettre de la vidéo à partir d'une caméra embarquée, il faut rajouter un transmetteur vidéo.

Paparazzi est reconnu pour ces qualités techniques et son coût relativement faible [11]. Aujourd'hui, il envisage d'améliorer son système de communication par l'ajout de la fonctionnalité de relais d'information et d'échange de communications inter-drones.

La section suivante présente des solutions de réseaux de communications permettant de communiquer les drones entre eux et avec la station de contrôle.

11. http ://www.objectifnews.com/node/478 Publié en 2009

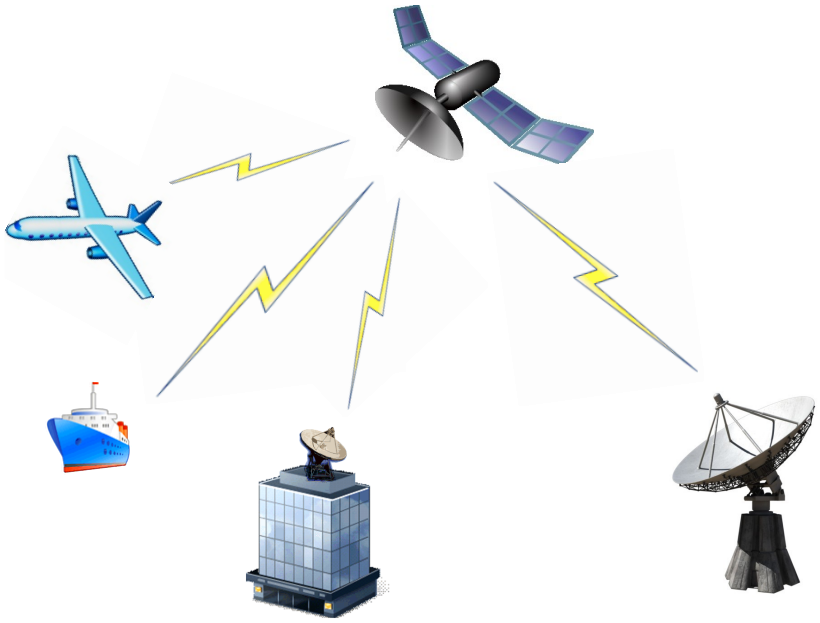


FIGURE 1.4 – Communications satellites

1.2 Solutions pour les réseaux de communications

Grâce au progrès dans le domaine des télécommunications, on vit aujourd'hui dans un monde connecté grâce aux différents types de technologies disponibles. En effet, il est possible de faire communiquer deux extrémités distant à l'aide d'une ou plusieurs technologies sans fil. Ainsi, pour faire communiquer plusieurs drones entre eux et avec leur(s) station(s) de contrôle, plusieurs possibilités existent comme les communications satellites, cellulaires ou ad hoc.

1.2.1 Réseaux satellite

Afin de faire communiquer deux points très éloignés l'un de l'autre, situés dans des régions où il est impossible d'avoir une infrastructure fixe, les communications satellites sont la meilleure solution.

Il existe deux types de satellites qui assurent la communication. Le premier type est le satellite géostationnaire ; il reste fixe par rapport à une position géographique puisque il tourne autour de la terre avec la même vitesse que cette dernière. Ces satellites couvrent une zone géographique fixe bien

déterminée.
Le deuxième type de satellite est le satellite orbital qui, contrairement aux satellites stationnaires, couvre des zones géographiques différentes puisque il se déplace sur une orbite autour de la terre.

Ce type de communication est souvent utilisé pour la télévision, les communications aéronautiques, maritimes militaires et les missions du contrôle des différentes zones géographiques sur la surface de la terre, etc. (figure 1.4)

Les satellites peuvent assurer la communication aussi entre la station sol et le drone dans les systèmes mono-drone. Par conséquent, pour les systèmes multi-drones, chaque aéronef peut communiquer avec la station de base à travers un satellite. Ainsi, les communications entre drones peuvent aussi passer de la même manière. Cependant, cette approche a quelques point faibles comme par exemple l'importante latence de transmission et le coût de lacement du satellite. De plus, les drones ainsi que la station de contrôle doivent être sur la ligne de visée du satellite. En effet, pour certaines missions, des arbres ou des bâtiments peuvent être des obstacles contre le signal échangé entre les drones et leur satellite relais.

Par ailleurs, la performance des systèmes satellitaires est liée à la puissance d'émission des émetteurs au sol, ce qui peut être un inconvénient pour les mini ou micro-drones qui sont équipés avec des batteries à faible capacité.

1.2.2 Réseaux cellulaires

Un autre type de communication, le plus utilisé de nos jours, est les réseaux de communications cellulaires (figure 1.5). Basée sur une topologie centralisée, cette technologie consiste à découper un territoire en zone (cellules), chacune est desservie par une station de base (le point central). Toute les communications doivent passer par ce point central qui a pour rôle de les acheminer à leurs destinations.

Les communications cellulaires sont la base des technologies de la téléphonie mobile comme le GSM, GPRS, UMTS, LTE [Mis04] et les communications de données sans fil comme le Wi-Fi et le WiMAX ([BGL08], [Ete08]) puisque ils offrent une extrême liberté pour les utilisateurs nomades.

Contrairement aux réseaux satellitaires, les réseaux cellulaires utilisent des émetteurs de faible puissance. Pour cela, il peuvent être une solution pour faire communiquer les drones en utilisant l'infrastructure des opérateurs téléphoniques déjà existante pour éliminer les contrainte de la portée et de la mobilité.

Cependant, le coût de communication n'est pas négligeable dans ce cas d'utilisation, même avec l'installation d'une nouvelle infrastructure. En plus,

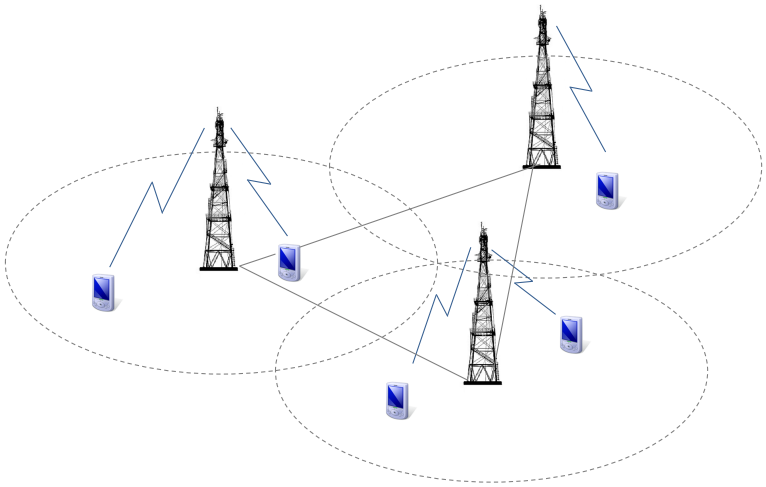


FIGURE 1.5 – Communications cellulaires

il est difficile de couvrir toute les zones et de garder cette infrastructure dans certains cas comme après les catastrophes naturelles.

1.2.3 Réseaux Ad hoc

Contrairement aux réseaux satellites ou cellulaires, les réseaux ad hoc mobiles (MANET pour *Mobile Ad hoc NETwork*) ne nécessitent pas une infrastructure fixe (des antennes relais ou satellite) pour acheminer les messages d'un nœud vers un autre (figure 1.6). Le principe des réseaux ad hoc est basé sur la coopération entre les différents nœuds du réseau. En effet, chaque nœud communique directement avec ses voisins qui se chargent de retransmettre les messages jusqu'à leur destination. Chaque nœud est un relais qui permet de retransmettre les paquets à leurs destination finale.

Cette coopération permet aux nœuds de bouger librement ce qui peut causer des changements fréquents et rapides de la topologie du réseau. Un changement qui peut causer la coupure de lien et en même temps la création de nouveaux . Ainsi, pour faire face aux changements dynamiques, fréquents et rapides de la topologie, le système ad hoc nécessite des protocoles de communications spécifiques puisque c'est un système autonome qui a le pouvoir de s'organiser automatiquement. Les protocoles de routage sont un exemple de ces protocoles. Ils sont responsables de détecter ce changement brusque et d'établir des chemins vers chaque destination. En effet, un chemin est la liste

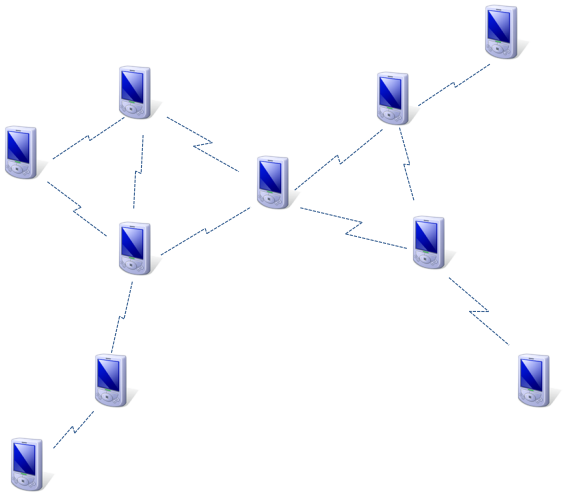


FIGURE 1.6 – Réseau Ad hoc

des nœuds intermédiaires (relais) à traverser par un paquet afin d'atteindre sa destination finale.

Par conséquent, les réseaux ad hoc sont bien adaptés aux systèmes de communication mobiles, dynamiques, ayant une topologie distribuée maillée, sans oublier leurs coûts raisonnable puisqu'ils ne nécessitent aucune infrastructure fixe.

1.2.4 Discussion

Dans les systèmes aéronautiques coopératifs sans pilote, plusieurs drones collaborent entre eux pour achever un but commun. Ces systèmes exigent une coordination totale entre leurs différents agents. Par conséquent, les aéronefs doivent avoir une idée actualisée en temps réel sur l'état et le déroulement des tâches des autres avions afin d'assurer la coordination entre eux et de pouvoir adapter leurs tâches à ce changement si nécessaire, comme par exemple le système TARF [TBV13] étudié pour les drones Paparazzi. Il s'agit d'une fonction dynamique permettant aux drones de s'auto-allouer de nouvelles tâches en cas d'événements soudains comme l'atterrissage d'urgence de l'un des aéronefs.

Par conséquent, les drones doivent échanger régulièrement des messages entre eux dans un environnement mobile dynamique. Pour cela, il faut choisir le système de communication le plus adapté aux caractéristiques des systèmes de mini-drones coopératifs.

Le tableau 1.1 compare l'utilisation des différents systèmes de communication présentés dans les sections précédentes dans le contexte de flotte de drones.

Système de communication	Avantages et/ou Inconvénients sur les réseaux de mini-drones
Communications satellite	– Latence d'émission – Coût élevé – Puissance d'émission importante
Communications cellulaires	– Coût élevé – Indisponibilité de l'infrastructure dans des situations critiques
Communication ad hoc	– Coût faible – Moins de contraintes sur la mobilité

TABLE 1.1 – Avantages ou inconvénients des systèmes de communication

Les mini-drones sont caractérisés par un poids léger, une capacité de charge limitée, et une capacité en énergie restreinte. En effet, un micro-drone ne peut pas supporter trop de charge puisque plus le matériel est léger plus l'engin peut atteindre de hautes altitudes et plus son endurance est longue [CYCG07].

Par ailleurs, les drones sont conçus pour être utilisés dans les endroits les plus dangereux et les plus difficiles à atteindre par l'homme. Il est difficile d'avoir une couverture cellulaire dans ces endroits à cause de la complexité d'implémentation des relais fixes dans ces lieux. De plus, après les catastrophes naturelles, ces infrastructures risquent d'être endommagés. En outre, il ne faut pas oublier le coût élevé de cette infrastructure et de sa maintenance ou même le coût de l'utilisation des relais déjà implémentés par les opérateurs de téléphonie mobile.

Par conséquent, les réseaux ad hoc sont bien adaptés pour le réseau de communications d'une flotte de drones.

1.3 Contributions

Le réseau ad hoc est une solution raisonnable pour faire communiquer les drones entre eux, à l'intérieur d'une flotte, et avec la station sol tout en respectant leur liberté de mouvement. De plus, les réseaux ad hoc sont rentables par rapport aux autres réseaux qui nécessitent une infrastructure

couteuse qui, dans certaines situations, risque d'être endommagées.

Pour les systèmes multi-agents, où différentes entités collaborent ensemble pour atteindre un but commun, plusieurs tâches complémentaires sont attribuées à chaque agent. Ces tâches ont différents niveaux d'importance ou de priorité et chacune d'eux nécessite des informations spécifiques et offre des résultats différents des autres. En appliquant ce genre de paradigme sur une flotte de drones, chaque aéronef aura une tâche d'un niveau d'importance ou de priorité différents des autres. En conséquence, les drones échangeront une variété de types de messages selon leurs résultats et leurs besoins. Ainsi, chaque drone aura des demandes spécifiques en termes de communication.

Par exemple, dans le cadre de missions de contrôle d'une zone géographique, une flotte de drones collabore en contrôlant chacun une petite section. Ces drones transmettent de la vidéo ainsi que la température mesurée à la station de contrôle. La température qui est une information envoyée régulièrement est moins gourmande en ressources que la vidéo. De plus, le drone qui s'occupe de la section la plus dangereuse doit envoyer une vidéo avec une meilleure qualité que les autres, ainsi ce trafic nécessite plus de débit. Ces besoins peuvent évoluer au cours du temps, puisque les tâches de chaque drone peuvent changer à n'importe quel instant.

Le système de communication utilisé pour ce genre de mission doit être conscient de ces besoins ainsi que de leur évolution au cours du temps. Il doit être capable de différencier le service et fournir à chaque classe de trafic la qualité de service demandée. Nous proposons dans cette thèse une architecture de communication conçue pour répondre à ces exigences. Cette architecture est adaptée aux réseaux des flottes de drones coopératifs. Elle est étudiée pour pouvoir réaliser ce genre de système avec Paparazzi. En effet, elle est implémentée sur des drones réels Paparazzi et testée dans un environnement réel.

Avant son intégration au système Paparazzi, les performances de cette architecture de communication sont évaluées par simulation en utilisant le simulateur de réseau OMNET++. Cette simulation est réalisée dans un environnement proche de la réalité afin de prédire les problèmes qui peuvent affecter le système dans son environnement réel. Pour cette raison, un modèle de mobilité reproduisant les mouvements réels de drones Paparazzi est crée.

L'objectif de cette étude est de créer un nouveau système de communication Paparazzi permettant de contrôler une flotte de drone coopératifs tout en respectant les différents besoins en terme de communication pour chaque aéronef afin de garantir les meilleures performances pour les opérations.

1.4 Structure du mémoire

Ce mémoire est organisé en six chapitres. Le premier présente le travail réalisé durant cette thèse ainsi que son cadre d'application et de déploiement.

Le deuxième introduit les réseaux ad hoc de drones et présente ces caractéristiques par rapport aux autres classes des MANET. Par la suite, il présente les différents techniques et protocoles abordés dans ce cadre d'études dans le but de pouvoir choisir les protocoles MAC et routage qui répondent à nos exigences.

Ensuite, il présente les différents mécanismes de gestion de la qualité de service (QdS) dans les réseaux filaires et ad hoc utilisés au niveau MAC et routage ainsi que des structures de différenciation de service.

La suite de ce chapitre présente les deux moyens que nous allons utiliser pour évaluer l'architecture de communication proposée : la simulation et l'expérimentation réelle. Pour cette raison, il présente les différents modèles de mobilité utilisés pour les simulations des systèmes de communication ad hoc dans le but de créer un environnement de simulation réaliste. Ensuite, il introduit des plateformes de tests des flottes de drones déjà réalisées.

Le troisième chapitre résume les exigences des systèmes de drones coopératifs et présente, par la suite, l'architecture de communication que nous proposons dans ce mémoire et détaille le fonctionnement de chacun de ses modules.

Le quatrième chapitre est dédié à l'étude d'évaluation des performances de l'architecture de communication proposée par simulation. Il est organisée en deux sections. La première section présente le modèle de mobilité que nous avons créé dans le but de reproduire les mouvements de drones Paparazzi et construire un environnement de simulation proche de la réalité. Nous détaillons le principe de ce modèle ainsi que l'étude réalisée pour valider son fonctionnement. Par la suite, la deuxième section aborde l'étude par simulation réalisée pour le système de communication proposé : elle présente les différents scénarios et analyse leurs résultats.

Dans le cinquième chapitre, les expérimentations réelles réalisées sont abordées. Ces expérimentations permettent d'évaluer et valider le fonctionnement de notre proposition dans son environnement de déploiement réel. D'abord, nous détaillons les différentes modifications matérielles et logicielles réalisées pour le système Paparzzi afin de pouvoir déployer notre architecture de communication. Par la suite, nous détaillons les scénarios des différents tests réalisés et nous analysons les résultats obtenus.

Le dernier chapitre conclue ce mémoire. Il résume le travail effectué ainsi que les contributions. Il est suivi d'un ensemble de perspectives pouvant permettre de compléter ce travail et d'explorer de nouvelles idées prometteuses

pour les domaines des réseaux de communication et des drones.

Conclusion

Afin de réussir une mission conçue pour une flotte de drones coopératifs, un échange de données continue est nécessaire permettant d'atteindre un niveau de coordination élevé entre les aéronefs et leur station de contrôle. Par conséquent, la communication joue un rôle important dans la réussite de ces opérations. Les réseaux ad hoc sont une solution prometteuse pour les communications entre les drones et avec la station sol.

Au cours d'une opération collaborative, les drones jouent différents rôles et échangent une variété de messages demandant différents besoins en termes qualité de service (débit, délai, pertes, etc.). Par conséquent, un système de communication conscient de ces demandes doit être utilisé pour gérer les ressources dans le réseau.

Ce chapitre a introduit le contexte de cette thèse et a donné des exemples sur les missions conçues pour des flottes de drones civils. Ensuite, une discussion autour des différents systèmes de communication a été présentée afin de trouver le système approprié. Finalement, notre contribution a été introduite. Le chapitre suivant présente les différentes propositions étudiées pour faire communiquer les drones à l'intérieure d'une flotte.

Chapitre 2

Réseaux ad hoc de drones

Contents

2.1 Taxonomie des MANET 20
2.1.1 Les VANET 20
2.1.2 Les AANET 21
2.1.3 Les réseaux Ad hoc de drones 21
2.2 Mécanismes pour les réseaux ad hoc de drones 25
2.2.1 Niveau MAC 25
2.2.2 Niveau routage 30
2.2.3 Mécanismes de gestion de la QdS 39
2.3 Évaluation des réseaux ad hoc de drones 51
2.3.1 Simulation 51
2.3.2 Expérimentation réelle 59

Introduction

Les réseaux ad hoc présentent une solution pour les communications entre les drones à l'intérieur d'une flotte et avec la station de contrôle. En effet, l'échanger de données entre les différents nœuds dans un réseau ad hoc, ne nécessite pas un élément central responsable de l'acheminement des communications comme le satellite ou les antennes relais utilisées dans les réseaux cellulaires. Il existe différentes classes de MANET classifiées selon des caractéristiques en termes d'utilisations et de déploiement. Par conséquent, chaque classe a ses propres exigences et besoins en termes de communication.

Ce chapitre présente les différentes classes de MANET et compare leurs caractéristiques dans la première section. Par la suite, il se focalise sur les

réseaux ad hoc de drones et présente les différents mécanismes étudiés pour permettre aux drones d'échanger des données à travers un réseau ad hoc. La troisième section de ce chapitre présente deux moyens d'évaluation des performances des systèmes de communication qui sont la simulation et l'expérimentation réelle.

2.1 Taxonomie des MANET

Les MANET sont adaptés aux réseaux distribués moyennement dense qui permettent de définir des routes entre les différents nœuds. Chaque système peut être caractérisé par des propres critères comme les conditions de son environnement d'application (densité, interférence, changement de topologie, etc), la capacité des équipements utilisés (portée radio, autonomie, énergie, etc). Par conséquent, il existe plusieurs sous-classes de MANET classifiées selon leurs utilisations, objectifs, déploiements ou types de communications. Les réseaux ad hoc véhiculaires VANET (*Vehicular Ad hoc NETwork*) ainsi que les réseaux ad hoc aéronautiques AANET (*Aeronautical Ad hoc NETwork*) sont deux classes de MANET.

2.1.1 Les VANET

Les VANET sont les réseaux de communications ad hoc au sein d'un groupe de véhicules dans le but de communiquer entre eux et avec des équipements fixes à portée (figure 2.1). Ce qui caractérise cette classe de

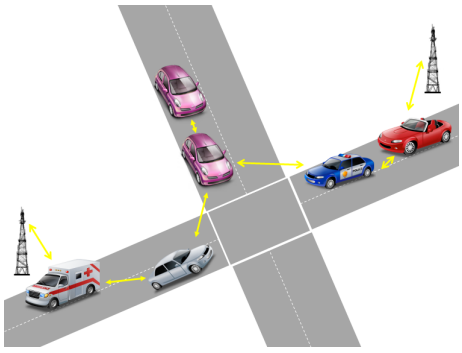


FIGURE 2.1 – Réseau Ad hoc véhiculaire

réseau par rapport aux autres est la mobilité des nœuds. En effet, les véhicules tendent à se déplacer ensemble d'une manière organisée suivant des routes bien définies. Ce mouvement est loin d'être au hasard, puisque le choix des

trajectoires pour atteindre une position géographique est limité. Il dépend du nombre des chemins disponibles.

Un autre critère spécifique des VANET est l'énergie. Contrairement aux autres équipements des MANET, les véhicules n'ont aucune contrainte sur l'énergie.

Les VANET sont la clef des systèmes véhiculaires du futur, dans lesquels les véhicules communiquent entre eux pour fournir des informations concernant la situation du trafic (comme l'état de la circulation, les travaux sur les routes, les accidents, etc.) aux conducteurs et aux autorités concernées.

2.1.2 Les AANET

Les réseaux ad hoc aéronautiques sont présentés comme une solution pour augmenter la sécurité des vols puisque les systèmes de communication sol-bord existant comme les communications satellites ou bande L sont limités en termes de capacité, de couverture et de coût de déploiement. En effet, l'utilisation des avions comme des relais permet d'étendre la portée des avions et de pouvoir propager ses données jusqu'une station sol [Bes13].

Ce système peut être utile pour des services pour les passagers ou pour les compagnies aériennes comme l'enregistrement de la boite noires en temps réel.

Les AANET et les VANET se ressemblent beaucoup en ce qui concerne la mobilité et l'énergie puisque les avions aussi suivent des trajectoires bien définies entre chaque point de départ et point d'arrivée et n'ont pas de contrainte d'énergie. Ces deux classes ont des différences au niveau de la vitesse et l'altitude. En effet, contrairement aux véhicules qui circulent sur le sol, les avions suivent des chemins dans différents niveaux d'altitude et avec des vitesses beaucoup plus importantes.
En se basant sur ces ressemblances, des chercheurs trouvent que les AANETs et VANET appartiennent à la même catégorie comme en [RPG13].

.

2.1.3 Les réseaux Ad hoc de drones

Les UAANET (*UAV Ad hoc NETwork*), les réseaux ad hoc de drones définit une nouvelle forme de MANET où les nœuds sont des drones qui permettent de retransmettre les messages vers leurs destinations. En effet, récemment des recherches ont été menées permettant de créer des systèmes multi-drones collaboratifs comme dans [MRM11] ou dans [BVH07] où plusieurs drones coopèrent entre eux afin d'accomplir une mission avec des meilleures performances. Ces systèmes collaboratifs nécessitent l'établissement

de communications inter-drones pour assurer la coordination entre les différents agents du système comme l'échange de positions géographiques pour l'évitement de collisions ou pour la re-planification autonome de tâches entre les agents d'une mission.

Les réseaux ad hoc de drones peuvent être appliqués aux systèmes où plusieurs drones sont utilisés, un seul drone ne peut pas créer un réseau ad hoc.

2.1.3.1 Comparaison réseaux de capteurs et UAANET

i. Rappel sur les réseaux de capteurs

Le réseaux de capteurs, WSN (*Wireless Sensor Network*), est un groupe de micro-capteurs qui contrôlent et collectent des informations sur les conditions de différents endroits comme par exemple la température, l'humidité, la pollution, le sens et la direction du vent.

Le succès de ces systèmes est dû au faible coût des capteurs, leur petite taille et leur efficacité dans plusieurs domaines puisque il en existe plusieurs types. Par contre, les WSN sont très limités en énergie ce qui nécessite l'utilisation de protocoles de communication économes en énergie.

ii. La comparaison

Dans la littérature, un autre nom désignant les réseaux ad hoc de drones est utilisé. Il s'agit du *réseau de capteurs aériens (Aerial Sensor Network)* comme dans [PZ09], dans [QKWS+10] ou dans [GHWL13]. Cependant, cette application est différente des systèmes multi-drones collaboratifs puisque c'est un réseau de capteurs mobiles ou les capteurs sont embarqués dans des drones. Ils collectent les données et les retransmettent vers un point de collecte, souvent au sol. La différence entre ce système et les UAANET se résume en trois point (tableau 2.1).

- Premièrement, le rôle des drones est différent dans les deux cas d'utilisation puisque dans les réseaux de capteurs aériens, le drone a un rôle passif, il n'est qu'un simple transporteur de capteur, alors que pour les UAANET les drones sont des agents du système, acteurs de leurs propres tâches et programmés selon des fonctions complexes afin d'assurer la coordination du système et même de prendre des décisions. Ainsi, ils ont leurs propres besoins d'information et de communication ;
- Deuxièmement, le sens du trafic est l'une des différences entre les réseaux de capteurs aériens et les UAANET puisque dans le premier système le trafic passe uniquement des capteurs vers le point de collecte tandis que dans un UAANET le trafic peut passer dans tous les sens : entre

les drones, des drones vers la station sol et de la station sol vers les drones, ce qui crée un réseau plus dynamique et plus chargé ;
– Le troisième point de différence est la densité. En effet, la densité de nœuds dans les réseaux de capteurs traditionnel est beaucoup plus importante que la densité des drones dans les systèmes multi-drones.

Critères	Réseau de capteur aérien	UAANET
Rôle des drones	passif	actif
Sens du trafic	un seul sens	plusieurs sens
Densité	+++	+
Énergie	- - -	-

TABLE 2.1 – Comparaison entre réseau de capteurs aériens et un réseau ad hoc de drones pour les systèmes multi-drones

Comparaison VANET et UAANET

Les UAANET sont une classe des MANET, elle hérite de ses caractéristiques de conceptions et diffère des autres classes par certaines spécificités. En effet, en comparant UAANET avec VANET on trouve plusieurs différences :

– **Mobilité et changement de topologie** : contrairement aux véhicules qui se déplacent ensemble suivant des chemins prédéterminés (des routes, des autoroutes ou des rails), les drones ont la liberté de bouger librement selon leurs plans de vol et parfois selon l'évolution de leurs tâches. Il est, donc, difficile de prédire les mouvements d'un drone puisque même avec un plan de vol prédéterminé, il y a toujours des changements imprévus à cause des conditions de l'environnement ou de l'évolution de la mission. En outre, chaque drone peut avoir son propre plan de vol modifiable selon les exigences de la mission ce qui cause le changement soudain, fréquent et parfois rapide de la topologie du réseau entrainant la perturbation de connections à cause de la rupture soudaine et la création des liens entre les aéronefs [YKB11]. Ce critère est la différence majeure entre UAANET et les autres MANET ;
– **Altitude et ligne de vision** : dans les VANET, les nœuds bougent proche du sol, pratiquement à la même hauteur. Par conséquent, il est difficile d'avoir une ligne de vision directe entre chaque émetteur et récepteur. Par contre, puisque les drones volent à une certaine altitude, il est possible d'avoir très souvent une ligne de vision entre les aéronefs

[BST13] ;
- **Énergie** : les UAANET, comme la majorité des classes de MANET, ont des sources d'énergie très limitée, ainsi ces systèmes nécessitent des protocoles de communications qui économisent de l'énergie afin d'augmenter leurs durées de vie. Cette contrainte est valide uniquement pour les petit drones. Cependant, dans les VANET, le réseau ne nécessite pas des protocoles économisant de l'énergie puisque les communications entre les engins sont supportés par les batteries des véhicules, donc ils n'ont aucun problème d'autonomie ;
- **Densité du réseau** : il est possible de trouver une dizaine de voitures par route, et quelques centaines de véhicules dans les auto-routes ce qui explique que les VANET sont en général des réseaux très denses contrairement aux UAANET où le système multi-drone comporte uniquement quelques drones.

Caractéristique	VANET	UAANET
Mobilité en groupe	oui	non
Type du mouvement	organisé suivant des routes pré-définies	aléatoire selon l'environnement et l'évolution de la mission
Vitesse	rapide	moins rapide
Source d'énergie	suffisante	limitée
Ligne de vision	rarement	plus souvent
Densité du réseau	importante	faible

TABLE 2.2 – Comparaison VANET et UAANET

Une autre exigence des UAANET est le passage à l'échelle. En effet, la collaboration entre plusieurs drones permet d'améliorer les performances du système par rapport aux systèmes mono-drone. Pour certaines missions, l'amélioration des performances du système est liée au nombre d'aéronefs impliqués. Par exemple, les auteurs dans [YCBE10] montrent que plus que le nombre de drones impliqués dans des missions de recherche et de sauvetage est grand, plus la mission est rapide.

Une fois appliqué, ce système permettra aux utilisateurs de drones de créer de nouvelles applications très innovantes.

2.2 Mécanismes pour les réseaux ad hoc de drones

Pour les missions de drones collaboratifs, la communication entre les différents engins joue un rôle très important afin d'assurer la coordination totale et garantir une meilleure performance. Le système de communication pour ce genre de mission doit faire face à un autre défi, il s'agit d'acheminer chaque message à sa destination avec la qualité demandée en termes de délais, de pertes et de débit.

Cette section présente les techniques et les protocoles de communication déjà utilisés et proposés pour assurer la communication dans les UAANET.

2.2.1 Niveau MAC

Le réseau ad hoc est un système auto-organisé d'engins connectés à travers des liaisons sans fil. Dans ce genre de réseaux, le canal de transmission radio est strictement contrôlé à cause de la bande passante disponible limitée par rapport aux réseaux filaires. De plus, d'autres facteurs peuvent réduire la capacité de la bande passante comme l'interférence et la congestion sans oublier que le support de communication sans fil peut être sujet à des erreurs lors de l'émission des signaux. Pour cette raison, un contrôle spécial et strict par des protocoles de contrôle d'accès au support (MAC pour *Medium Access Control*) est nécessaire. Le niveau MAC est responsable de certains mécanismes très spécifiques comme la création des trames, l'adressage, le contrôle d'erreur et la résolution des conflits d'accès au canal entre les différents nœuds du réseau. Cette section introduit les protocoles les plus utilisés pour les MANET. Par la suite, elle étudie des proposition pour les UAANET.

2.2.1.1 Dans les MANET

Malgré les spécificités d'un UAANET par rapport aux autres classes des MANET, il partage avec eux plusieurs similitudes en ce qui concerne les exigences de conception puisque finalement ils sont tous des sous-classes des MANET. En effet, il est possible d'utiliser pour les UAANET la norme IEEE 802.11 (Wi-Fi).

Le Wi-Fi est un standard de communication sans fil utilisé pour les MANET puisque il peut opérer en deux modes : le mode ad hoc (figure 2.2 (b)) et le mode centralisé (figure 2.2 (a)) où les nœuds échangent des informations à travers un ou plusieurs équipements centraux, appelés point d'accès.

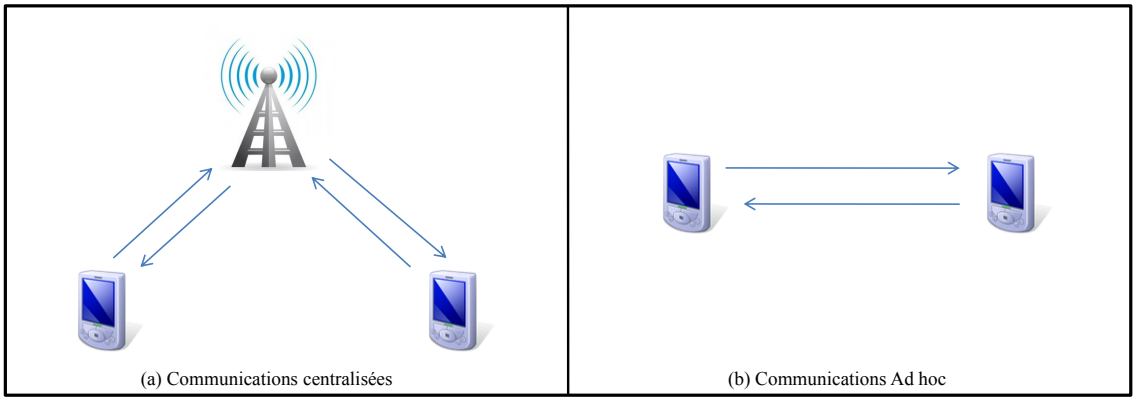


FIGURE 2.2 – Communications ad hoc et communications centralisées

Pour avoir accès au canal, deux fonctions peuvent être utilisées : la fonction de coordination par point PCF (*Point Coordination Function*) utilisée pour les réseaux centralisées et la fonction de coordination distribuée DCF (*Distributed Coordination Function*) qui est plus flexible puisqu'elle peut être utilisée sur n'importe quelle architecture de réseau.

DCF est basée sur le mécanisme CSMA/CA (*Carrier Sense Multiple Access* avec *Collision Avoidance*). Avec ce mécanisme, chaque nœud doit écouter le canal avant d'émettre afin d'éviter les émissions simultanées de plusieurs stations ce qui conduit à des collisions. Le CSMA/CA est très efficace dans le cas de voisins cachés qui sont des nœuds du même réseau situés hors de portée radio de l'autre qui, par conséquent, ne peuvent pas détecter leurs émissions. Ainsi, avant d'émettre les donnés, chaque émetteur et récepteur utilisant le mécanisme CSMA/CA doivent échanger quelques messages afin d'informer les voisins de la transmission à venir comme illustré par la figure 2.3.

L'émetteur doit initialiser le dialogue par l'envoi d'une requête d'émission RTS (*Request To Send*) qui est une requête d'accès au canal. Recevant le message RTS, l'émetteur répond avec le message CTS (*Clear To Send*) indiquant que le nœud est prêt à recevoir les données. Les autres voisins écoutant ces deux messages ou l'un d'eux, selon leurs positions, décalent leurs émissions puisqu'ils sont au courant que le canal est occupé. Après cet échange, les données peuvent être envoyées en toute sécurité sans aucun risque de collision.

Malgré la popularité du standard 802.11 dans les réseaux ad hoc, il présente quelques inconvénients puisqu'il ne supporte pas la QdS. En effet, il ne comporte pas de mécanismes de différentiation de service qui permettent de garantir la bande passante nécessaire pour les trafics les plus prioritaires.

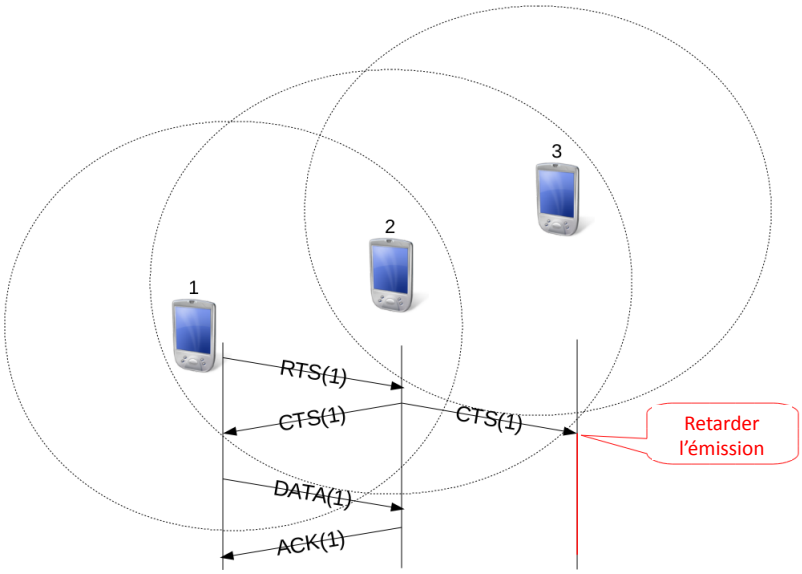


FIGURE 2.3 – Le mécanisme CSMA/CA

Ces mécanismes peuvent être essentiels dans un réseau comme UAANET où différents types de messages sont échangés.

Il existe plusieurs versions du Wi-Fi avec des améliorations au niveau des performances. Tout a commencé avec le standard IEEE 802.11b qui fournit un débit entre 5.5 et 11 Mbits/s. Cette version est très utilisée pour les UAANET. En effet, elle a été utilisée dans [TBH04], [KLHH05] et [MMN+]. Une nouvelle amélioration du Wi-Fi a donné naissance au standard IEEE 802.11a qui offre un débit théorique égal à 54 Mbits/s. Il a été utilisé pour un réseau UAANET dans [CHKV06].

Plusieurs autres protocoles sont utilisés pour les réseaux MANET comme, Bluetooth [Blu], HiperLAN [Kru92], ou encore MH-TRACE (*Multi-Hop Time Reservation using Adaptive Control for Energy Efficiency*) [TH05] présenté ultérieurement.

2.2.1.2 Dans les UAANET

Pour les communications des drones avec la station sol, d'autres standards peuvent être utilisés comme le IEEE 802.15.4 , un des réseaux sans fil personnel WPAN (*Wireless Personnal Area Network*). Il définit le niveau physique et MAC du Zigbee, utilisé dans plusieurs applications du réseau de capteurs et aussi dans le système Paparazzi pour connecter le drone à

la station de contrôle. L'inconvénient majeur de ce protocole est qu'il a un débit très faible (250 kbits/s) par rapport au Wi-Fi. Par conséquent, il ne supporte pas les trafics temps réel comme les données multimédia malgré sa capacité à établir des connexions à travers une longue distance.

Dans [AD10a], un nouveau protocole MAC conçu pour les UAANET est proposé, il est nommé AMUAV (*Adaptive MAC protocol for UAV*). Ce protocole est conçu pour être utilisé sur des drones équipés par deux types d'antennes : directionnelle et omni-directionnelle. En effet, les antennes directionnelles ont plusieurs avantages par rapport aux antennes omnidirectionnelles puisqu'elles rayonnent avec une puissance d'émission plus importante dans une seule direction ce qui permet d'optimiser l'utilisation de l'espace, d'améliorer la fiabilité de transmission des données, d'étendre la portée de transmission ainsi que d'économiser la consommation d'énergie [BJ12]. Le rôle d'AMUAV est d'organiser l'utilisation des antennes selon le type des paquets envoyés : les paquets de contrôle (RTS, CTS et ACK) sont émis à travers l'antenne omnidirectionnelle alors que les paquets de données sont envoyés par l'antenne directionnelle.

Malgré l'amélioration qu'offre cette proposition pour le débit de transmission, le délai et le taux d'erreur dans les systèmes multi-drones, elle reste couteuse puisque basiquement les drones sont équipés d'un seul type d'antenne.

Une autre proposition pour utiliser des antennes directionnelles est détaillée dans [TB13], nommée LODMAC (*Location Oriented Directional MAC protocol*). Cette proposition est basée sur la découverte des voisins et elle est conçue pour être utilisée dans des réseaux de drones avec une plateforme à haute altitude HAP (*High altitude platform*) qui est un aéronef quasi-stationnaire à une altitude stratosphérique (jusqu'à 25 km) permettant de fournir des services de communication ou de supervision pour une durée de quelques jours à quelques semaines. Par rapport au systèmes au sol ou satellite, un HAP a l'avantage d'être facile à déployer et d'avoir une large couverture permettant une connexion avec une ligne de vision directe.

La proposition de [TB13] est étudiée dans le scénario dans lequel le HAP survole un réseau de drones pour pouvoir communiquer directement avec chaque aéronef du réseau qui échange du trafic avec les autres drones et la station de contrôle au sol à travers un réseau ad hoc (figure 2.4).

LODMAC, est adapté à ce scénario. En effet, il nécessite trois antennes directionnelles pour chaque drone. La première est utilisée pour la découverte des voisins à travers les communications avec le HAP. La deuxième est utilisée pour échanger des message de contrôle comme RTS ou CTS. La troisième est utilisée pour les émissions de données. Les trois antennes sont synchronisées et gérées par le protocole LODMAC proposé.

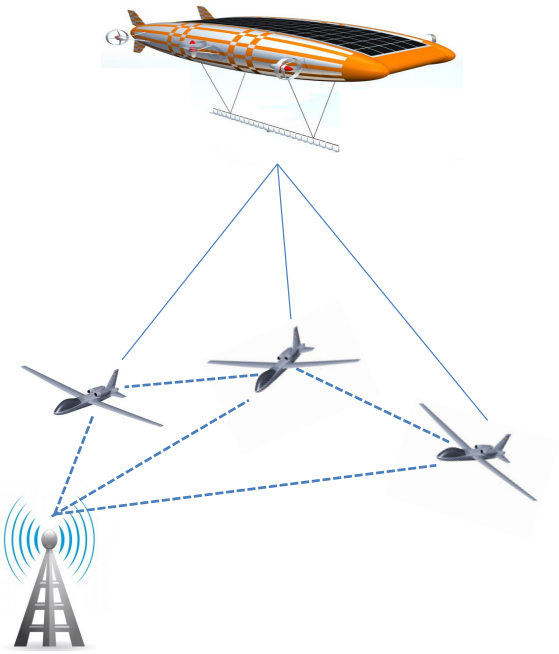


FIGURE 2.4 – Le scénario de LODMAC

Dans [CYL+13], un autre protocole MAC pour les UAANET basé sur le principe des transmissions sans fil en duplex intégral (Full-duplex) est présenté. En effet, les réceptions et les émissions dans les liaisons sans fil ne peuvent pas être simultanées, soit l'une, soit l'autre. En plus, la réception des signaux de différents émetteurs ne permet pas de recevoir correctement les messages à cause des collisions et d'interférence. Cependant, des recherches ont rendu possible la réception de plusieurs signaux de différents émetteurs MPR (*Multi-packet reception*) en utilisant le standard IEEE 802.15.4 [CJS+10], [LZH12].

Le principe du protocole MAC proposé dans [CYL+13] est de permettre aux drones d'avoir des communications sans fil en duplex intégral et en MPR afin d'échanger fréquemment des messages de contrôle contenant des informations sur l'état du canal CSI (*Channel State Information*). Les simulations ont montré l'efficacité de cette proposition sur les performances du réseau.

Pour notre cas, Nous devons choisir un protocole MAC qui ne doit pas être couteux puisque le système Paparazzi est un système libre utilisant des matériels pas couteux. De plus, la capacité de son canal doit satisfaire les besoins du trafic à échanger. Pour ces raisons, nous optons pour le Wi-Fi.

La section suivante présente les protocoles de routage proposés pour les

UAANET.

2.2.2 Niveau routage

Afin d'acheminer une information entre deux drones distants, un ou plusieurs nœuds intermédiaires dans le réseau coopèrent entre eux par la retransmission des données jusqu'à leurs destinations. En effet, les protocoles de routage sont responsables de choisir la meilleure route vers la destination qui minimise les collisions, les interférences, ou encore le délai.

Il existe une grande diversité de protocoles de routage. Plusieurs protocoles se basent sur les adresses IP pour désigner les nœuds alors que d'autres se basent sur leurs positions géographiques.
Différentes taxonomies sont proposées pour classifier les protocoles de routage comme dans [LZH+06]. La classification traditionnelle des protocoles de routage est basée sur la manière de découvrir la topologie du réseau. Elle définit plusieurs classes, les plus populaires sont les réactifs, les pro-actifs, les hybrides et les géographiques.

Les protocoles présentés dans cette section sont proposés pour les réseaux ad hoc de drones.

2.2.2.1 Protocoles de routage utilisant l'adressage IP

***i.* Les protocoles de routage réactifs**

Ceux sont les protocoles qui établissent des routes uniquement à la demande. Un nœud désirant communiquer avec un autre, cherche une route dans sa table de routage. S'il en trouve une, il passe directement à l'émission, sinon il initie une phase de découverte de route. Une route vers une destination est maintenue jusqu'à ce que la destination devienne inaccessible ou jusqu'à ce que la route soit expirée, c'est à dire elle n'est plus utilisée depuis une certaine période.

Le principal inconvénient de ce type de protocole est leur réaction lente face au changement de topologie.

AODV

AODV (*Ad hoc On-Demand Distance Vector*) [PR99a] est un protocole de routage réactif.

Il utilise trois messages de contrôle pour déterminer les routes.

- RREQ (*Route Request*) : la requête envoyée pour demander des itinéraires ;
- RREP (*Route Reply*) : la réponse aux RREQ ;
- RERR (*Route Error*) : le message d'erreur utilisé pour signaler que une route n'est plus disponible.

AODV-DTN

AODV est la base de plusieurs autres protocoles conçus pour différentes classes de réseau y compris les UAANET. En effet, dans [LPG06], les auteurs présentent une combinaison de AODV et un protocole de routage tolérant au délai DTN (*Delay Tolerant Network*) pour un système hybride drones et nœuds au sol.
Un réseau DTN est un réseau qui supporte les anomalies temporaires de communication et les latences de plusieurs minutes. Son principe est de *stocker et transmettre* (*Store and Forward*). En effet, les routeurs, dans un réseau DTN, se permettent de garder les paquets temporairement quand ils ne trouvent pas de solution pour les router, jusqu'à l'établissement d'une nouvelle route. Ces réseaux permettent de limiter le taux de perte du trafic échangé.

AODV-DTN est utilisé dans des systèmes ayant plusieurs drones qui assurent la connexion entre des groupes disjoints de nœuds mobiles au sol. Sa stratégie de routage est d'utiliser AODV comme le protocole sous-jacent avec un léger support du DTN afin de créer un protocole cohérent adapté à ce genre de systèmes. En effet, AODV est utilisé au niveau sol pour router les messages des nœuds mobiles des différents groupes au sol alors que le routage DTN est appliqué entre les drones qui se permettent de garder les paquets jusqu'à l'établissement des routes vers les destinations suite aux mouvements des aéronefs.
Ce protocole est dédié aux systèmes qui tolèrent les délais, ainsi, il ne peut pas être utilisé pour échanger des trafics temps-réel (comme les flux interactifs).

Time-slotted AODV

Dans [FHS07], une autre proposition de protocole basée sur AODV est présentée, (*A Time-Slotted On-Demand Routing Protocol*) définie pour les réseaux de drones. Ce protocole a pour but l'augmentation de la fiabilité des communications puisqu'il est proposé pour être utilisé avec la fonctionnalité complexe de formation de vol autonome. Cette fonctionnalité nécessite une collaboration entre les drones par l'échange des données entre eux.

Time-slotted AODV permet d'affecter des créneaux temporels pour chaque nœud afin d'émettre ses paquets vers son voisin. L'évaluation de ce protocole montre que, par rapport à AODV, il permet de réduire le taux de perte des paquets dues aux collisions ce qui augmente la fiabilité du système.

***ii.* Protocoles de routage pro-actifs**

Contrairement aux protocoles réactifs, les protocoles pro-actifs déterminent les routes périodiquement et indépendamment du trafic. Ce type de protocole évalue les routes au sein du réseau d'une manière continue en maintenant ses

données concernant les itinéraires vers chaque nœud du réseau, cohérentes et à jour. Par conséquent, le chemin de n'importe quel paquet à transmettre est déjà connu et peut être utilisé immédiatement. En effet, tous les nœuds stockent des informations de routage qui sont mises à jour à chaque changement dans la topologie du réseau.

Le principal inconvénient de cette catégorie est la quantité importante d'informations utilisées pour la maintenance des tables des routages.

OLSR

OLSR (*Optimized Link State Routing Protocol*) [MCL+01] est un exemple des protocoles pro-actif. Il maintient les routes dans une table mise à jour à travers l'émission périodique des paquets «*Hello*». En effet, OLSR utilise deux types de messages :

- le message «*Hello*» : utilisé pour détecter les voisins directs, situés à la portée de communication de l'émetteur. Il contient la liste des voisins connus. Ce message est périodiquement retransmis vers les voisins des voisins (voisins à deux-sauts) ;
- le message de contrôle de la topologie est utilisé périodiquement pour la maintenance des informations du système récoltées.

Cet échange permet à chaque nœud de calculer des routes vers chaque destination bien qu'il surcharge le réseau.

Afin de réduire l'interférence causée par ce fonctionnement, la méthode MPR (*Multi-Point Relays*) est utilisée. Les MPRs sont des nœuds sélectionnés pour retransmettre les messages vers leurs voisins afin de réduire le nombre de retransmission dans le réseau. En plus ils ont un rôle principal dans le routage et la sélection des chemins entre chaque source et sa destination. En général, les MPRs sont les voisins à un saut qui couvrent le maximum des voisins à deux sauts (figure 2.5).

DOLSR

Comme AODV, OLSR est la base d'autres propositions comme DOLSR (*Directional Optimized Link State Routing*) présenté dans [AD10b]. Ce protocole est conçu pour être utilisé avec des antennes directionnelles et en inter-couches (*cross layer*). La technique d'inter-couches permet de communiquer entre deux couches qui ne sont pas voisines ainsi que de lire et de contrôler les paramètres des autres couches. Par conséquent, elle permet de concevoir conjointement plusieurs protocoles de différents niveaux. Cette approche est considérée comme une méthode efficace pour améliorer les performances d'un réseau sans fil.

Le but de DOLSR est de diminuer le nombre des MPRs afin d'améliorer

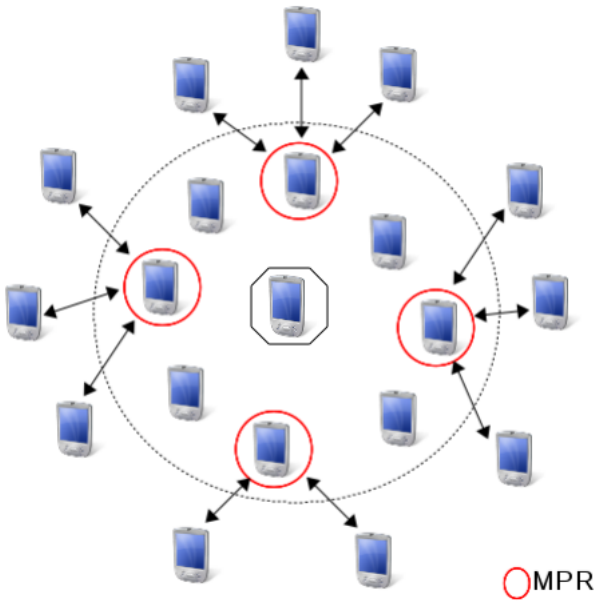


FIGURE 2.5 – Les MPRs

les performances des réseaux ad hoc entre les drones en termes de délais de bout en bout, trafic de contrôle et d'interférence.

Nous remarquons que la majorité des protocoles de routage réactifs et proactifs proposés pour les UAANET sont des descendants des deux protocoles AODV et OLSR. Cela est causé par le nombre limité des études réalisées pour cette classe de réseau jusqu'à présent.

2.2.2.2 Protocoles de routage géographiques

Les protocoles de routage géographiques sont des protocoles qui font références aux nœuds dans le réseau par leurs postions géographiques et non pas par leurs adresses IP. Ces positions sont supposées être connues par tous les nœuds grâce à un échange périodique d'informations entre eux permettant de créer des tables de localisation des voisins. Par conséquent, un émetteur peut choisir un relais, le plus proche de la destination, pour acheminer les paquets en se basant sur des critères géographiques comme la distance euclidienne (dans la transmission géographique gourmande «*greedy geographic forwarding*»).

L'application des protocoles géographiques en réalité semble être complexe puisque c'est compliqué de connaitre les positions géographiques de

tous les autres nœuds du réseau par un échange de données ou même par l'utilisation d'un serveur centralisé responsable de diffuser des données de localisions. En réalité, dans un réseau mobile, il est difficile de garantir que tous les nœuds restent à l'intérieur de la portée radio du serveur ou encore qu'ils reçoivent tous les échanges des autres voisins sans oublier la complexité du calcul de la position géographique.

GPSR

GPSR (*Greedy Perimeter Stateless Routing*) [KK00] est un exemple de ces protocoles. Il permet de marquer les paquets avec la positon géographique de leurs destinations et par la suite chaque nœud émetteur peut faire un choix optimal de son prochain saut en se basant sur la méthode gloutonne (*greedy*) : le voisin ayant la position géographique la plus proche de celle de la destination finale du paquet comme dans la figure 2.6.

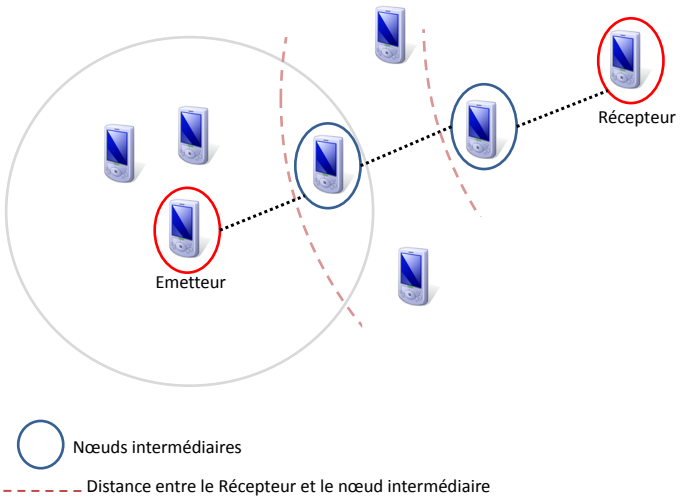


FIGURE 2.6 – La transmission géographique *gloutonne*

Dans le cas où un nœud échoue à trouver un relais optimal vers la destination, GPSR se rétablit par le routage autour du périmètre de la région.

Dans [HBM07], une étude comparative par simulation de trois protocoles de routage : un géographique (GPSR), un réactif (AODV) et un pro-actif (OLSR), dans le contexte d'un réseau ad hoc de drones, montre que GPSR offre de meilleures performances que les autres protocoles en termes de taux de livraison et de délai des paquets échangés.

En utilisant GPSR, les auteurs de [LMB09] proposent de bénéficier des échanges des positions géographiques réalisés par le protocole de routage

pour améliorer les performances de la mission de recherche. Ce protocole, nommé USMP (*UAV Search Mission Protocol*), se base sur les informations géographiques de routage pour résoudre le conflit entre les trajectoires des aéronefs. USMP permet d'améliorer l'efficacité de la recherche en terme de couverture et de minimiser les changements de direction des drones.

GPMOR

Dans [LSLY12], un nouveau protocole de routage géographique a été proposé pour les réseaux ad hoc de drones hautement dynamique. Contrairement à la méthode classique qui se base sur les informations de localisation des nœuds, cette proposition, nommée GPMOR (*Geographic Position Mobility Oriented Routing*), essaye en plus de prédire le mouvement des drones qui bougent suivant le modèle de mobilité *Gauss-Markov* [CBD02]. En effet, chaque drone est supposé conscient de sa localisation géographique grâce au système GPS. Il échange périodiquement sa position avec ses voisins directs uniquement. Étant donné que la période entre deux échanges est fixée à quelques secondes afin de diminuer l'interférence, les aéronefs essayent de prédire le mouvement de leurs voisins et de définir leurs nouvelles positions durant cet intervalle de temps. Par conséquent, il est possible de sélectionner le relais optimal vers la destination qui peut, elle même, changer de position entre temps. La comparaison de ce protocole avec d'autres protocoles de routages géographiques comme GPSR montre que GPMOR est meilleur en termes de latence et de taux de livraison de paquets puisque le système d'adressage utilisé pour router les paquets (les positions géographiques) est plus fiable.

LAROD

Un autre protocole de routage géographique est proposé dans [KNT08]. Ce protocole, nommé LAROD (*Location Aware Routing for Opportunistic Delay Tolerant*), est un protocole de routage tolérant au délai. LAROD a été évalué par simulation dans le contexte d'un scénario réaliste d'une mission de reconnaissance collaborative. Il a pour but la découverte du chemin le plus court vers chaque destination en choisissant un ou plusieurs relais. En effet, l'ensemble des voisins qui garantissent un minimum de progrès vers la destination sont des relais potentiels.
Chaque drone émetteur diffuse les paquets vers ses relais potentiels qui déclenchent des minuteurs de durées choisies aléatoirement. Seul le premier aéronef ayant le minuteur expiré peut rediffuser le paquet vers ses voisins. Les autres, écoutant la transmission, retirent leurs copies du paquet. Dans le cas où un drone n'arrive pas à trouver un relais permettant de garantir un progrès

vers le drone cible, il réagit selon le principe DTN défini précédemment. Les paquets restent stockés temporairement jusqu'à ce que la mobilité de l'aéronef crée d'autres chemins. Une fois que le paquet a atteint sa destination, celle-ci répond par un acquittement afin d'empêcher les transmissions indéfinies des paquets entre les nœuds.

Ce protocole n'est pas robuste puisque en cas où un nœud échoue pour une raison quelconque, tous les paquets stockés dans ce nœud vont être perdus à moins d'avoir d'autres copies dupliquées dans d'autres nœuds.

2.2.2.3 Protocoles de routage hybrides

Les protocoles de routage hybrides combinent différents types de mécanismes. Ils peuvent bénéficier ainsi des avantages de chacun.

RGR

L'article [SSHK+11] présente une étude de performances de GPSR dans le contexte du réseau ad hoc de drones. Cette étude montre que dans le cas d'un réseau ou les aéronefs sont dispersés, GPSR rencontre des problèmes de fiabilité. Pour cette raison, les mêmes auteurs ont proposé dans [Shi11] un protocole de routage hybride comme une solution pour la défaillance de GPSR dans le contexte de réseaux de drones coopératifs. Il s'agit d'une combinaison entre le principe de routage géographique gourmand (*greedy geographic forwarding*) et le protocole de routage réactif AODV. Ce protocole est nommé RGR (*Reactive-Greedy-Reactive routing protocol*).

Chaque drone émetteur établit une route vers la destination en se basant sur la méthode réactive de AODV. En cas où un nœud de cette route ne parvient pas à trouver un chemin à cause de la mobilité, il passe à la méthode géographique (figure 2.7). Le paquet est acheminé ainsi vers le voisin le plus proche géographiquement de la destination finale.

Le passage de l'adressage IP à l'adressage géographique semble compliqué puisque chaque catégorie nécessite un échange spécifique de données et de différents mécanismes de traitement et de calcul.

2.2.2.4 Protocole de routage hiérarchique

Un autre type de protocoles qui peut être efficace pour les UAANET est la famille des protocoles hiérarchiques qui permet d'améliorer le passage à l'échelle dans le réseau. Ce type de protocole se base sur la répartition des nœuds en des groupes (*cluster*) selon des critères comme la mission, l'altitude ou la zone géographique. Chaque groupe élit un nœud comme tête de groupe

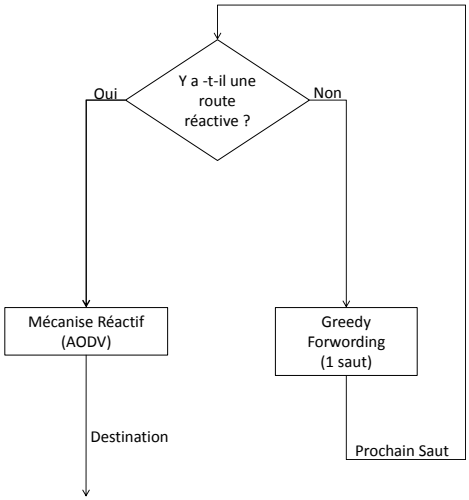


FIGURE 2.7 – Mécanisme du protocole RGR

TG (*Cluster Head*) qui doit être connecté directement aux autres membres du groupe et il est responsable de la communication avec les autres groupes à travers leurs TGs (figure 2.8)

Ce genre de protocole peut être hybride en combinant différents types de mécanismes pour router les informations à l'intérieur d'un groupe (entre les nœuds du même groupe) et à l'extérieur des groupes (entre les différents TGs). EHSR [XHG+01] (*The Extended Hierarchical State Routing*) est un exemple de protocole hiérarchique hybride. Il est utilisé dans des scénarios où les drones sont des relais pour des nœuds terrestres (figure 2.9). Dans ce cas les nœuds sont groupés selon leurs altitudes : les drones forment un premier groupe et les nœuds terrestres forment les autres groupes comme présenté dans la figure 2.9.

Deux méthodes de routage sont utilisées pour router les paquets à l'intérieur et à l'extérieur des groupes. En effet, pour communiquer entre les différents groupes, le mécanisme de routage de vecteur de distance est utilisé alors que pour les communications locales, inter-groupe, le routage à état de lien est appliqué.

La différence entre le routage de vecteur de distance et le routage à état de lien est que le premier choisi la meilleure route en fonction de la distance qui définie le nombre de sauts vers la destination, alors que le deuxième sélectionne sa meilleure route en se basant sur la qualité de chaque lien sur le chemin. Ces deux types présentent une autre classification de protocoles de routage basée sur la manière de sélectionner la meilleure route.

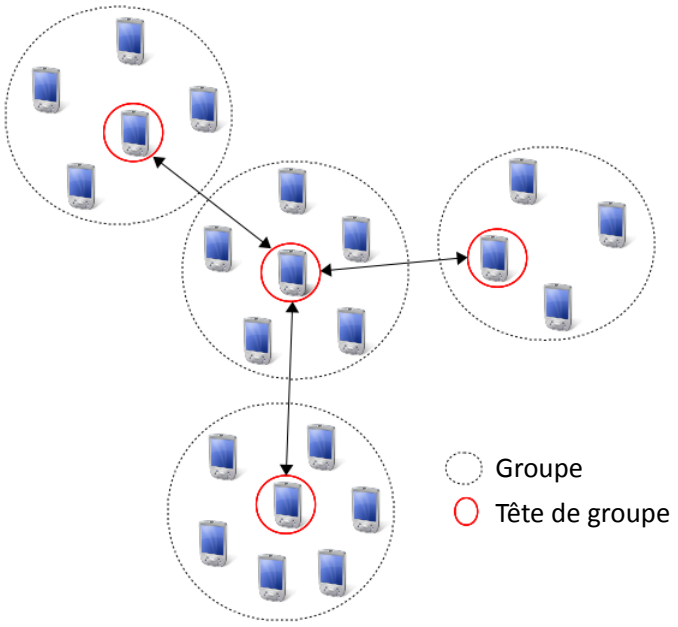


FIGURE 2.8 – Méthode de groupge

Discussion

Comme présenté dans le tableau 2.3, chaque protocole de routage étudié dans cette section a une spécificité. En effet, la majorité, est étudiée pour être appliquée dans des conditions spécifiques comme EHSR ou AODV-DTN qui nécessitent une architecture hybride entre des drones et d'autres groupes de nœuds au sol, ou encore DOLSR qui nécessite deux types d'antenne.
De plus, l'application réelle de certains autres protocoles semble très complexe comme RGR qui combine deux types d'adressage ou GPMOR qui se base sur la prédiction des mouvements des nœuds selon la loi de Gauss-Markov. Ce protocole n'est pas applicable avec d'autres modèle de mobilité aléatoire. De plus, dans la réalité, les mouvements de drones peuvent être différents du modèle de mobilité Gauss-Markov.

Dans notre étude, nous avons besoin d'un protocole de routage qui consomme le minimum de bande passante et qui n'utilise pas des fonctionnalités complexes puisque nous envisageons d'implémenter notre architecture sur des vrais drones Paparazzi. Par conséquent, AODV est le protocole approprié pour nos exigences puisqu'il est réactif et implémenté sur Linux ainsi

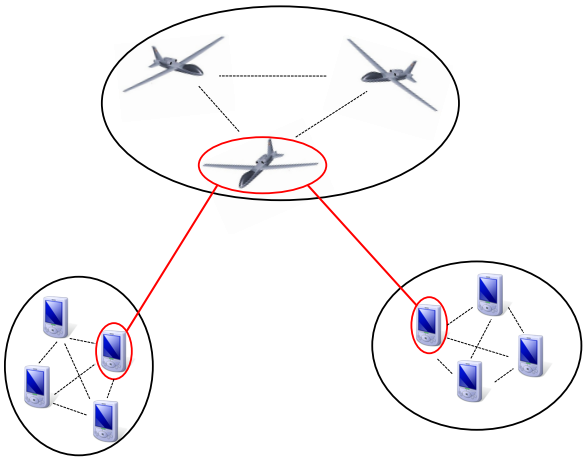


FIGURE 2.9 – Architecture hybride utilisée avec le protocole EHSR

que la majorité des simulateurs réseau. Nous optons pour l'implémentation AODV-UU [GSB09] car elle est disponible pour la simulation ainsi que pour le système Linux.

Cette section a présenté les protocoles de routage étudiés pour les réseaux ad hoc de drones. La section suivante se focalise sur la qualité de service, l'une des exigences des UAANET.

2.2.3 Mécanismes de gestion de la QdS

La qualité de service (QdS) est une évaluation du comportement du réseau en terme des performances de services offerts comme le délai de bout en bout, le taux de perte, le débit ou encore la variation de délai (la gigue). Elle représente les exigences du service qui doivent être garanties par le réseau.

De plus, avec un grand nombre d'utilisateurs qui utilisent une variété d'applications, des mécanismes de gestion de la QdS permettent de différentier les services et d'améliorer le comportement du réseau selon les besoins de chaque application. Par exemple, les applications temps réel comme le trafic interactif et les vidéos conférences nécessitent plus de débit et moins de délai que d'autre type d'applications comme l'échange d'e-mail ou le transfert de fichiers.

Dans les réseaux ad hoc de drones, la QdS est un élément primordial pour garantir de meilleures performances des missions puisque au cours d'une opération, les types de messages échangés entre les drones diffèrent selon le rôle que joue l'aéronef dans l'évolution de la situation et selon les équipements

Classe	Protocole	Référence	Commentaires
Réactive	AODV	[PR99a]	Implémentation réelle
	AODV-DTN	[LPG06]	Architecture de réseau hybride
	Time-slotted AODV	[FHS07]	Solution fiable pour missions de formation de vol
Pro-active	OLSR	[MCL+01]	Implémentation réelle
	DOLSR	[AD10b]	Utilisation de deux types d'antennes
Géographique	GPSR	[KK00]	Réalisation complexe
	GPMOR	[LSLY12]	Applicatiopn avec le modèle de mobilité Gauss-Markov
	LAROD	[KNT08]	Protocole non robuste pour les opérations de recherche et reconnaissance
Hybride	RGR	[Shi11]	Complexité de la combinaison entre adressage IP et géographique
	EHSR	[XHG+01]	Architecture de réseaux hybride

TABLE 2.3 – Les protocoles de routage pour les UAANET

et les capteurs qui y sont embarqués. En effet, les drones équipés par des caméras, qui réalisent des tâches de surveillance ou de recherche, émettant du trafic temps réel nécessitent plus de débit et moins de délai que les drones équipés par des thermomètres qui sont sensibles aux pertes et demandent un maximum de fiabilité.

Par conséquent, le système de communication doit être capable de respecter les exigences de chaque message et de chaque service dans le réseau et de partager équitablement les ressources entre les drones.

La différentiation du trafic et l'assurance de la QdS sont les responsabilités des différents niveaux du modèle OSI. Cette section présente différents mécanismes de gestion de la QdS étudiés pour les réseaux filaires, MANET et UAANET

2.2.3.1 Dans les réseaux filaires

i. Couche transport

Plusieurs études ont été réalisées autour de la création de protocoles et de mécanismes qui permettent d'améliorer la QdS au niveau transport. Le protocole TCP (*Transmission Control Protocol*) [TCP81], est un exemple de

ces protocoles. Il a été conçu pour garantir une communication de bout en bout fiable. Le protocole XTP (*Xpress Transfer Protocol*) [XTP92] est un autre protocole de transmission fiable conçu pour les applications temps-réel et à haut-débit comme le transfert du trafic interactif. Il fournit différentes fonctionnalités comme le contrôle de trafic, le contrôle de débit, le contrôle d'erreur et l'ordonnancement des messages à priorités différentes.

De plus, le protocole SCTP (*Stream Control Transmission Protocol*) [RS07] est un protocole de transmission fiable conçu pour transporter la signalisation du réseau téléphonique commuté (*Public Switched Telephone Network (PSTN)*) sur IP. Il offre la fiabilité, la remise en ordre des séquences, et le contrôle de congestion. Il existe aussi un protocole de transmission pour les trafics temps-réel qui ont des contraintes sur les délais. Il s'agit du protocole RTP (*Real-time Transport Protocol*). Il est utilisé principalement pour les transmissions voix sur IP ou vidéo conférence. RTP fonctionne en mode unidirectionnel ce qui permet de limiter la consommation des ressources réseau dans le but d'augmenter considérablement le débit obtenu par les applications utilisant ce protocole.

ii. Couche réseau

Afin de pouvoir créer différents types de service autre que le service *au mieux* (Best-effort), qui est le service basique du protocole internet (IP) permettant de transporter les données sans aucune garantie sur la fiabilité ou sur les performances du trafic, deux approches sont utilisées : l'ingénierie de trafic et la gestion de files d'attentes. L'ingénierie de trafic comprend différentes méthodes et techniques pour optimiser les ressources et gérer le trafic dans le réseau en choisissant, vers chaque destination, la meilleure route qui garantie la QdS demandée par le service. Le protocole MPLS [RAV01] (*Multi-Protocol Label Switching*) est l'un des protocoles utilisés dans l'ingénierie de trafic. Il combine l'intelligence du routage IP avec les performances de la commutation.

MPLS permet de créer un chemin d'accès spécifique pour chaque séquence de données identifiée par une étiquette (*label*) collée sur chaque paquet. Cette méthode permet d'économiser ainsi le temps nécessaire pour un routeur afin de rechercher l'adresse du nœud suivant sur la route.

iii. IntServ et DiffServ

Pour pouvoir différencier les services, il faut utiliser des mécanismes de gestion des files d'attente pour chaque paquet. Ces mécanismes sont : la mise en file d'attente (*Queuing*) qui gère le temps d'attente dans la file et l'ordonnancement (*Scheduling*) qui est le processus permettant de donner

accès aux ressources. La méthode FIFO (*premier entré premier servi «First In First Out»*) est l'algorithme le plus simple de gestion de file d'attente.

Il existe des modèles de fournitures de QdS combinant plusieurs mecansimes de gestion de QdS dans le but de différencier le service et améliorer la QdS pour chaque classe comme IntServ [BCS94] (*Integrated Services*) et DiffServ [BBC+98] (*Differentiated Services*). Ces deux structures ont été standardisées au sein de l'IETF (*Internet Engineering Task Force*) [1].

IntServ est la première architecture proposée pour améliorer la QdS pour l'Internet. Elle permet de gérer des flux entiers de données et a pour but de garantir les besoins du trafic temps-réel. Ces flux, appelés micro-flux, sont identifiés par le protocole utilisé (UDP ou TCP) et leurs adresses et numéros de port source et destination.

Avec IntServ, les ressources sont réservées au niveau de chaque routeur sur le chemin pour chaque flux de trafic afin de fournir une garantie de bout en bout. La réservation de ressource est assurée par une signalisation de bout en bout gérée par le protocole RSVP [BEB+97] (*Resources Setup Reservation Protocol*).

RSVP et IntServ sont basés sur un mécanisme (*Soft State*). Ce mécanisme consiste à garder une idée sur l'état du réseau au niveau de chaque routeur qui gère un contrôleur d'admission permettant d'accepter ou de rejeter la demande du service selon les données sur l'état actuel du réseau.

DiffServ est un modèle de QdS qui permet de réserver les ressources à l'avance sans échanger de signalisation en remplaçant la granularité *par flux* d'IntServ par la granularité *par agrégats*. En effet, DiffServ définit plusieurs catégories de classe de service. Chaque classe de service définie sa façon de traitement demandée en terme d'un ensemble de paramètres (bande passante, délai, etc) pour le domaine DiffServ. Le domaine DiffServ comporte deux types de routeurs :

- Les routeurs d'extrémité (*Edge Routers*) : ils sont situés sur les bords du domaine (les premiers à entrer en contact avec les flux), et ils sont chargés de traitements complexes comme la classification et le marquage ainsi que le mise en forme du trafic ;
- Les routeurs de cœur (*Core Routers*) : les routeurs intermédiaires dans le réseau. Ils ont un rôle plus simple qui consiste à acheminer des paquets selon leurs marquages.

Les routeurs traitent chaque paquet selon son marquage. Pour cela ils utilisent une variété de stratégies de planification et de gestion de file d'attente

1. http ://www.ietf.org/

au cours de l'acheminement des paquets vers leurs destinations définissant le comportement PHB (*Per Hop Behavior*). Le PHB propose trois classes de service :

- *Expedited Forwarding* (EF) : la classe de trafics qui ont des contraintes sur les délais. Elle est idéale pour les flux interactifs ;
- *Assured Forwarding* (AG) : l'ensemble de trafics qui exigent une garantie sur la livraison ;
- Le trafic ordinaire ou *Best-effort*.

2.2.3.2 Dans les MANET

Dans les MANET, la mission de garantir la QdS demandée est plus délicate que dans les réseaux filaires à cause des ressources limitées du canal sans fil et la courte durée de vie du lien entre les nœuds. En effet, à cause de la mobilité des nœuds, la topologie du réseau change fréquemment générant la création de nouveau liens et la rupture d'autres soudainement. Ce phénomène peut être la cause d'une fluctuation du délai de transmission. De plus, les nœuds dans un MANET agissent en même temps comme un terminal sans fil ordinaire qui peut recevoir et émettre des messages et comme un routeur en retransmettant les messages des autres émetteurs vers ses voisins. Pour cela, ce genre de réseau est plus complexe qu'un simple réseau filaire notamment en terme de congestion. Par conséquent, les mécanismes de fourniture de la QdS conçus pour les réseaux filaires comme IntServ ou DiffServ ne sont pas adaptés aux MANET.

Plusieurs propositions ont été étudiées afin d'améliorer la QdS dans ce genre de réseau dans les différents niveaux.

Les protocoles présentés dans cette section sont proposés pour les MANET en général. Comme par exemple l'étude réalisée dans [AHAN05] au niveau transport. Elle permet de résumer une variété d'études réalisées pour améliorer les performances du protocole TCP dans les MANET. Elles proposent une solution des problèmes comme l'incapacité de TCP à connaitre la cause de la perte qui peut être un problème au niveau de la route, ou un problème de congestion du réseau.

i. Niveau réseau

Un protocole de routage à QdS a pour but l'amélioration des performances du réseau. Il permet de définir les routes en se basant sur des critères et des informations qui concernent la QdS (le débit ou le délai) comme dans [LCT+05] où un protocole de routage hybride appelé MAR (*Mobile Agent Routing*) est proposé. Il fournit une connectivité fiable dans des scénarios militaires en choisissant la meilleure route vers la destination en terme de la

QdS. Le problème majeur de ces protocoles est la consommation de la bande passante et l'interférence causée par l'échange des informations sur l'état des routes. Ces échanges ont pour but de maintenir une vision mise à jour de la qualité des liens dans le réseau qui change très souvent à cause du dynamisme des nœuds.

De plus, le niveau réseau permet de gérer les files d'attente dans le but de respecter les priorités entre les paquets émis.

Ce paragraphe présente des protocoles de routage permettant d'établir des routes en se basant sur des données sur la qualité des liens afin d'améliorer la QdS dans les MANET.

BPQMRP

Un protocole de routage permettant d'améliorer la QdS est présenté dans [QAKQ11] nommé PBQMRP (*Position Based QoS Multi-cast Routing Protocol*). Il s'agit d'un protocole de routage multi-cast qui permet de sélectionner la route qui offre la meilleure QdS en termes de bande passante et du délai d'un nœud source vers un ensemble de destinations. BPQMRP se base sur la méthode de répartition de nœuds en groupe *clustering* (figure 2.8) dans des zones hexagonales de mêmes dimensions. L'évaluation des performances de ce protocole montre qu'il permet d'améliorer le taux de perte dans le réseau et de diminuer l'interférence causée par les messages de contrôle.

AOMDV

Dans le même contexte, [MD01] présente une autre proposition de protocole de routage dans le but d'améliorer la QdS du réseau. Il s'appelle AOMDV (*Ad hoc Multi-path Distance Vector*) et il propose une modification du protocole réactif AODV. Il s'adapte aux réseaux dynamiques où les liens entre les nœuds ont une faible durée de vie causant les ruptures fréquentes et soudaines des routes. AOMDV est caractérisé par une récupération rapide et efficace face à la rupture brusque des liens et au changement rapide de la topologie puisque il permet de trouver plusieurs routes vers la destination au cours de la phase d'établissement de routes et contrairement à AODV, il garde toutes ces données dans sa table de routage. Par la suite, AOMDV trouve rapidement une alternative de la route échouée vers la destination et permet de parvenir à une amélioration remarquable en terme de délai de bout en bout et de réduire l'interférence causée par les messages de contrôle de routage.

CEDAR

De même, CEDAR [SSB99] (*Core-Extraction Distributed Ad hoc Routing*)

est un protocole réactif permettant d'améliorer la qualité de service en terme de bande passante. Il est basé sur la notion du cœur du réseau qui est un sous-réseau formé par un certain nombre de nœuds permettant de couvrir le reste du réseau. Afin de limiter les échanges circulant dans le réseau et consommant la bande passante, uniquement les nœuds du cœur du réseau transmettent les données de contrôle sur l'état des liens.

CEDAR est basé sur trois composantes essentielles :

- l'extraction du cœur du réseau : phase de sélection d'un groupe réduit de nœuds responsables de l'établissement des routes et leur maintenance ;
- propagation d'état de lien : informations à jour au niveau de chaque nœud du cœur. Ces informations contiennent la liste des liens stables, à forte bande passante ;
- calcul de route : Le plus court chemin qui satisfait la bande passante demandée est sélectionné en se basant sur les données enregistrées au niveau du nœud cœur correspondant au sous-réseau de l'émetteur.

CEDAR est un protocole qui n'offre pas une garantie sur la bande passante puisque il ne fait pas de la réservation de ressource. Il permet d'offrir une route stable vers les destinations qui risque d'être un peu compliqué dans le cas de grand réseau dynamique où le nombre de liens ayant une qualité très variante est important.

AODV-QoS

Un autre protocole réactif QdS est AODV QoS présenté dans [PR99b]. Cette proposition est une modification du protocole de routage AODV afin de garantir une meilleure QdS en termes de délai et de bande passante puisque elle consiste à rajouter une extension dans les messages de contrôle contenant des informations sur le lien.

Afin de garantir la bande passante demandée, la requête AODV *RREQ* transporte la valeur de la bande passante demandée. Chaque nœud recevant ce message compare la valeur transférée avec la capacité du lien. Si elle est plus grande, le paquet est détruit. Par conséquent, chaque nœud recevant la réponse *RREP* compare sa capacité avec la valeur transférée et garde le minimum des deux valeurs.

QOLSR

De même pour OLSR, une amélioration a été proposée en [MBAAP02] afin d'avoir un routage à base de QdS en ajoutant des extensions à ses messages de contrôle en termes de délai et de bande passante. De plus, QOLSR améliore aussi le concept des nœuds MPR puisqu'il en rajoute la fonctionna-

lité du contrôle d'admission. En effet, les MPRs sont responsables d'accepter ou de refuser des flux afin de garantir que le flux d'un nouvel utilisateur ne dégrade pas les flux existant. Pour faciliter cette tâche, une valeur seuil globale est fixée pour la consommation de la bande passante. En cas de saturation, aucun trafic n'est accepté jusqu'au déchargement du réseau.

ii. Niveau MAC

Un protocole MAC à QdS permet de fournir les propres besoins en terme de QdS pour chaque nœud dans le réseau partageant le même canal avec ses autres voisins, sans affecter les besoins des autres nœuds.

WAVE

Le standard IEEE 802.11p [WAKP08] est un exemple de protocole MAC à QdS, connu sous le nom de WAVE (*Wireless Access in Vehicular Environment*). Il utilise la fonction EDCA (*Enhanced Distributed Channel Access*) qui a pour principe la différentiation du trafic en des catégories avec des priorités différentes. Ce protocole a une importance remarquable pour les applications sensibles au délai comme la transmission de la voix ou les trafics interactifs.

MH-TRACE

De plus, [TH03] présente un protocole MAC QdS nommé MH-TRACE (*The Multi-Hop Time Reservation using Adaptive Control for Energy Efficiency*). Ce protocole combine les avantages du système distribué et ceux du système centralisé en adoptant la méthodologie de groupage de nœuds *clustering* (figure 2.8). MH-TRACE se base sur le principe de la communication sur des tranches de temps bien définies pour chaque émetteur (TDMA pour *Time Division Multiple Access*) afin de limiter les collisions entre les paquets émis par les têtes de groupe. TDMA est un mécanisme utilisé dans les réseaux GSM et satellite permettant à de multiples utilisateurs de partager la même fréquence du canal en répartissant le temps d'utilisation en un cycle de tranches (*time slot*). Chaque tranche est allouée à un émetteur. Chaque nœud émet durant sa propre tranche de temps, et entre, par la suite, dans une phase de veille pour le reste du cycle. L'accès au canal pour les nœuds du même groupe est géré et contrôlé par le nœud en tête de groupe qui a pour but de limiter le taux de collision entre les paquets. L'avantage majeur de MH-TRACE est qu'il fournit de la QdS aux applications temps-réel comme la transmission de la voix ou la conférence vidéo grâce à sa méthode de gestion de la durée d'émission et sa méthode coordonnée d'accès au canal. De plus, il permet d'améliorer l'efficacité énergétique du réseau puisque les

nœuds peuvent entrer en veille plus souvent. Le système MH-TRACE est exploité dans [TH05] afin de créer une architecture inter-couche (*cross layer*) pour des transmissions en multicast, MC-TRACE (*Multicasting Time Reservation using Adaptive Control for Energy Efficiency*). Cette architecture est conçue principalement pour la transmission de la voix afin de réduire la dissipation de l'énergie durant l'émission et le repos du nœud.

iii. Solutions de fourniture de la QdS

D'autre structures, conçues entre le niveau MAC et routage, sont proposées pour les réseaux ad hoc afin de différentier le trafic et de garantir la QdS. Ces modèles permettent de réserver les ressources nécessaires pour chaque flux prioritaire en se basant sur la signalisation.

En effet, la signalisation a pour but la réservation, le rafraîchissement et la libération des ressources dans le réseau. Le transfert de la signalisation doit être fiable et de haute priorité pour pouvoir garantir la QdS à tout moment. De plus, les messages de la signalisation ne doivent pas avoir un coût élevé permettant de consommer davantage de la bande passante.

Il existe deux types de signalisation : une signalisation *in-band*, qui utilise l'entête IP du paquet de données pour signaler les ressources demandées, et une signalisation *out-of-band* qui utilise des paquets de contrôle spécifiques. La signalisation out-of-band a un coût plus important ce qui consomme davantage des ressources dans le réseau. RSVP est un exemple de protocole de signalisation out-of-band.

Protocole de signalisation INSIGNIA

INSIGNIA [LC98], (*In-Band Signaling Support for QoS In Mobile Ad hoc Networks*), est le premier protocole de signalisation proposé pour les réseaux ad hoc. Il est basé sur le principe de la signalisation *in-band*.

Avec la collaboration d'un contrôleur d'admission, INSIGNIA est un support pour la QdS pour les services temps réel. Il permet aux paquets ayant des besoins en termes de QdS d'exprimer leurs demandes dans un champ spécifique ajouté à l'entête IP. En effet, chaque nœud désirant émettre un trafic exigé en termes de QdS, commence à envoyer les paquets de données en mode Best-effort jusqu'à la réception de la notification d'acceptation de la réservation. Ces paquets contiennent dans leurs en-têtes la demande de réservation.
La demande de réservation comporte la valeur de la bande passante désirée ainsi que la valeur de la bande passante minimale en dessous de laquelle le flux ne souhaite pas descendre. Chaque nœud intermédiaire recevant ces messages décide d'accepter la réservation soit avec la bande passante optimale,

minimale ou la dégrader en Best-effort. Au cours de l'émission des données, la réservation peut être modifiée : augmentée à la valeur optimale ou dégradée à la valeur minimale, en modifiant l'en-tête IP et elle est rafraichie avec le passage des paquets de données. Une fois la réservation de la route est établie, le champ ajouté à l'entête IP est utilisé pour échanger l'état des liens.

De l'autre coté, le nœud récepteur envoie régulièrement des rapports sur la qualité du lien permettant à l'émetteur de modifier son débit d'émission.

INSIGNIA est un protocole de signalisation qui nécessite d'être associé à un contrôleur d'admission pour pouvoir affecter les ressources aux flux prioritaires. Bien que le principe de INSIGNIA permette de limiter le coût de la signalisation afin d'établir, adapter et restaurer des réservations, il nécessite une modification de l'entête IP ce qui est une possibilité trop complexe à réaliser pour l'implémentation réelle que nous allons réaliser.

FQMM

Le premier modèle de QdS proposé pour MANET est le modèle FQMM [XYLC00] (*Flexible QoS Model for MANET*). Il combine à la fois les deux structures IntServ et DiffServ. Il s'appuie ainsi, sur un schéma hybride de la granularité *par flux* de IntServ pour gérer le trafic le plus prioritaire et *par classe* de DiffServ pour les autres trafics dans un réseau constitué au maximum d'une cinquantaine de nœuds mobiles, formant un domaine DiffServ. FQMM utilise les mécanises de IntServ pour réserver la bande passante nécessaire pour les flux prioritaires. La limitation de l'application de IntServ permet de conserver les ressources dans le réseau.
Selon le sens des trafics, les nœuds peuvent avoir des rôles différents comme dans DiffServ :

- Nœud d'entré (*Ingress Node*) : est le nœud source, il est le point d'entrée du trafic ;
- Nœuds de cœur (*Core node*) : sont les nœuds intermédiaires ;
- Nœud de sortie (*Egress Node*) : est le nœud destination.

La classification du trafic est réalisée au niveau des nœuds d'entrée alors que la QdS est fournie tout au long du trajet, au niveau de chaque nœud du cœur selon le PHB de chaque paquet (au niveau du champ de DiffServ dans l'entête IP) jusqu'à arriver au nœud destination.

Le but de FQMM est de pouvoir interfacer le réseau avec l'Internet puisqu'il utilise les mécanismes du réseau filaire.

SWAN

[GsAS02] présente un autre modèle de gestion de la QdS nommé, SWAN (*Stateless Wireless Ad hoc Network*). SWAN est une structure sans état

basée sur des algorithmes de contrôle distribués permettant d'assurer une différentiation des services dans les réseaux ad hoc en deux classes : Best-effort (sans contraintes sur la QdS) et temps-réel (trafic UDP avec contraintes sur la QdS). Il contrôle le trafic Best-effort afin de garantir la priorité totale au trafic temps réel.
Le contrôle d'admission est réalisé au niveau de la source. En effet, un message de signalisation traverse la route vers la destination permettant de sonder la bande passante disponible dans le réseau et retournant vers l'émetteur. Recevant un bilan sur la qualité des liens sur la route, la source prend la décision concernant l'admission de flux ou non.
Une fois un émetteur décide de la possibilité d'émettre un trafic temps-réel, chaque nœud intermédiaire doit faire passer ce trafic en priorité en contrôlant le trafic Best-effort.

En cas de congestion, les bits ECN (Explicit Congestion Notification) de l'entête IP sont marqués afin de permettre à la source de réinitialiser son mécanisme de contrôle d'admission. Si la route ne dispose pas d'assez de bande passante, le trafic est supprimé. SWAN utilise un classificateur permettant de différentier les deux classes de trafic et contrôler le débit du trafic Best-effort.

Un mécanisme de contrôle du trafic Best-effort n'est pas suffisant pour garantir la QdS demandée par les flux admis. En effet, les flux temps réel eux même peuvent causer la congestion du réseau et la dégradation de la qualité des autres trafics. De plus, il n'y a pas de limite d'admission de trafic temps-réel ce qui peut même causer la perte totale du trafic Best-effort générant un vrai problème puisque tous les messages de contrôle de routage et de la couche d'accès sont considérés dans cette approche comme du trafic Best-effort.

2.2.3.3 Dans les UAANET

Au cours d'une opération coopérative, les drones échangent différents types de messages avec une variété de priorités. En effet, des trafics de même protocole, ayant la même destination et les mêmes numéros de port peuvent avoir différents besoins en terme de QdS. De plus, les besoins d'un même trafic peuvent évoluer au cours du temps. Ces exigences sont dues au fait que les drones peuvent jouer différents rôles au cours de l'opération. Certains drones accomplissent des tâches plus critiques que les autres et nécessitent un traitement de données plus prioritaire ce qui justifie que les trafics échangés dans le réseau peuvent être traités différemment. En outre, la tâche d'un drone peut changer au cours du temps selon les circonstances de la mission. Par conséquent, ses besoins en termes de QdS peuvent varier aussi.

Quelques propositions sont étudiées spécialement pour les réseaux de

drones. En effet, [SFS+04a] propose un protocole de routage conscient des besoins QdS des drones. Il permet de définir des priorités entre les paquets selon le protocole et l'émetteur. Par la suite, ces priorités sont respectées au cours de l'acheminement des données ce qui permet d'améliorer le débit et le délai essentiellement pour le trafic le plus prioritaire.

Une autre proposition dans [BT11] permet d'améliorer la QdS selon la tâche effectuée pour le drone. Cette solution est étudiée dans le cas où une flotte de drones opère ensemble d'une manière synchronisée et change de tâche simultanément. Les passages d'une tâche à l'autre sont gérés et contrôlés par la station de contrôle.

Étant donné que l'architecture du réseau est hiérarchique, les drones sont organisés en groupes. Pour chaque groupe, le drone ayant la meilleure qualité de lien avec la station sol est choisi comme *SuperNode*, est responsable de l'acheminement du trafic entre les nœuds de ces groupes et la station sol. Cette proposition définit la liste des tâches à effectuer par la flotte de drones et les ordonne selon les besoins de chacune en terme de QdS : les tâche qui nécessitent des exigences élevées de QdS et d'autre qui nécessitent des exigences moins élevées.

Le système de gestion de la QdS est composé par le processus de gestion de motif (*Process Pattern Manager*) qui contient la liste des tâches et le gestionnaire de files d'attente.
Il peut être utilisé en deux modes :

- Au niveau des nœuds ordinaires (qui ne sont pas des SuperNode) : la gestion de QdS est appliquée pour les propres paquets du nœud (le trafic sortant) ;
- Au niveau du *SuperNode* : la gestion est appliquée pour les paquets retransmis (trafic routé) ;

Pour chaque paquet sortant, le gestionnaire des files d'attente consulte le gestionnaire des tâches pour déterminer le traitement nécessaire durant la tâche en cours et la file d'attente appropriée. Le champ ToS de l'entête IP est utilisé pour marquer les paquets. Au niveau du *SuperNode*, les paquets sont traités selon le marquage de leurs champs ToS.

Cette proposition permet d'améliorer les performances du réseau ad hoc de drones dans un cas spécifique où les tâches des aéronefs sont prédéfinies et respectées durant toute l'opération. Cependant, dans un cas plus dynamique où les drones sont plus interactifs et ont des tâches qui s'adaptent aux circonstances, le processus de pré-définition la liste des tâches sera plus complexe.

Dans [SFS+04b], les auteurs proposes un prototype d'une mission coopérative pour une flotte de drones consciente de la QdS disponible dans le réseau. En effet, les drones utilise un logiciel qui permet d'adapter la qualité

d'image envoyée selon l'état de la mission et du réseau.

La QdS est l'une des exigences des réseaux ad hoc de drones qui nécessite l'utilisation de certains mécanismes adaptés à ce genre de système. Par conséquent, des études dédiées doivent être réalisées permettant de garantir la Qds demandée dans un environnement réel.

Discussion

Ces différents modèles de fourniture de QdS (FQMM et SWAN) permettent de différencier le trafic globalement en deux catégories : temps-réel qui a la priorité absolue et le reste de trafic (Best-effort). Cette manière de gestion de QdS peut aboutir, dans des situations de congestion, à la pénalisation et la dégradation du trafic Best-effort. L'architecture de communication pour les UAANET doit être capable de fournir chaque trafic ses besoins en terme de QdS tout en préservant le trafic Best-effort.

Cette section a présenté les différents mécanismes utilisés afin de différentier les services d'un réseau et de garantir la QdS demandée dans des systèmes filaires et sans fil. Une étape très importante de la création de nouveaux systèmes, la validation et l'évaluation de performances, est présentée dans la section suivante.

2.3 Évaluation des réseaux ad hoc de drones

L'évaluation des performances d'un système de communication est une étape importante dans le processus de création de nouvelles propositions notamment dans le contexte des réseaux de drones. En effet, cette évaluation permet, non seulement d'évaluer les performances du système, mais aussi de prédire les problèmes possibles qui peuvent affecter le système une fois déployé dans un environnement réel. Par conséquent, plusieurs améliorations peuvent être étudiée pour le système afin de l'adapter aux conditions de l'environnement réel.

Plusieurs moyens permettent d'évaluer les systèmes de communication comme la simulation, l'émulation, la modélisation mathématique ou encore l'expérimentation réelle. Pour notre étude, nous optons pour la simulation ainsi que l'expérimentation réelle.

2.3.1 Simulation

La simulation est la modélisation du comportement du réseau dans un environnement virtuel. Les protocoles et les expérimentations sont définis dans des fichiers modifiables et sont reproductibles.

La simulation a une grande popularité dans les études de performance des systèmes de communication. Elle permet de créer, dans un monde virtuel, tous les composants de l'environnement réel comme les positions des nœuds, leurs mouvements, la zone et les protocoles de communications. Pour avoir une meilleure étude de performances des systèmes, il faut réaliser les simulation dans un environnement proche de la réalité. Le modèle de mobilité, le modèle de propagation des signaux, les atténuations sont des composants de l'environnement virtuel de la simulation. Dans notre étude, nous nous focalisons sur les modèles de mobilité qui font différer les réseaux ad hoc de drones des autres réseaux ad hoc.

2.3.1.1 Modèles de mobilité

Le modèle de mobilité définit les trajectoires des nœuds ainsi que leurs vitesses. Il permet au simulateur de calculer les positions géographiques des nœuds à chaque instant au cours de la simulation. Ces positions permettent de mesurer les distances entre les différents nœuds afin de définir les liaisons entre eux. De plus, dans le cas des réseaux ad hoc, la mobilité des nœuds permet de modifier fréquemment la topologie du réseau en causant la création de nouveaux liens et la rupture d'autres. En effet, en changeant de position, des nœuds se rapprochent d'autres nœuds et s'éloignent d'autres. En conséquence, de nouveaux liens se créent et d'autres sont détruits. Cette modification peut être fréquente, brusque et rapide causant la perturbation de la connexion.
Le modèle de mobilité permet de traduire ce changement de topologie dans l'environnent virtuel de la simulation.

Dans [CBD02], les auteurs ont réalisé une étude permettant de comparer le comportement du protocole de routage DSR en utilisant une variété de modèles de mobilité. Les résultats montrent que les performances de DSR en termes de délai, taux de livraison et interférence ont été fortement influencées par le changement du modèle de mobilité. Par conséquent, pour réaliser une meilleure étude de performances il faut choisir le modèle de mobilité le plus proche de la réalité des mouvements des nœuds. En effet, plus les mouvements des nœuds du réseau dans le modèle sont proches de la réalité, plus la simulation réseau a des chances d'aboutir à des résultats proches de la réalité et donc d'être utile. Ainsi, pour pouvoir évaluer un protocole de communication ad hoc de drones, il faut créer un modèle de mobilité réaliste.

i. Modèles de mobilité pour les MANET

Cette partie présente les modèles de mobilité les plus utilisés utilisé pour

les MANET. Elle fait l'objet d'un projet de Master réalisé dans le contexte de cette thèse.

Random Way-Point

Plusieurs chercheurs étaient conscients de la nécessité de l'utilisation d'un modèle de mobilité réaliste et ont proposé des modèles de mobilité proches de leurs cas d'utilisation. Cependant, le modèle le plus utilisé reste le *Random Way-Point* (RWP) [CBD02] qui se base sur des mouvements rectilignes vers des positions destinations choisis aléatoirement au cours de la simulation.

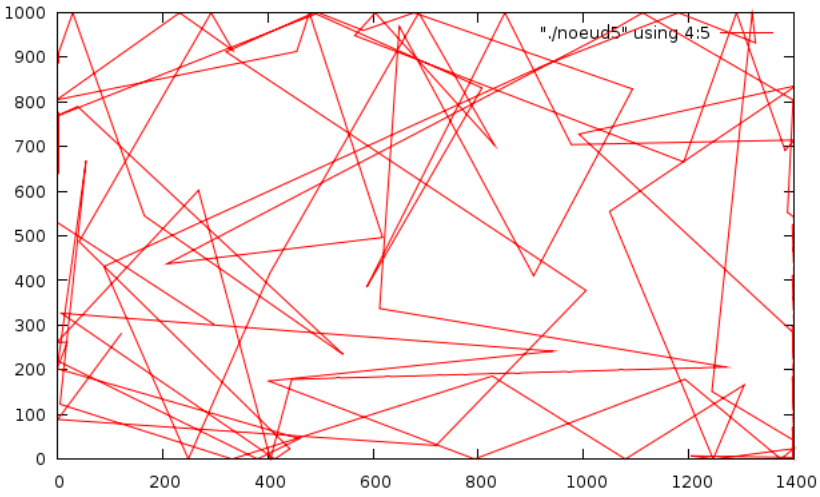


FIGURE 2.10 – Mouvements de RWP

Chaque nœud choisit aléatoirement une destination dans la surface de simulation et une vitesse. Ensuite, il se dirige vers cette destination suivant une trajectoire rectiligne avec la vitesse déjà choisie. Une fois arrivé à sa destination, le nœud s'arrête pour une durée de temps avant de réitérer le même processus. (figure 2.10)

Random Direction

Random Direction (RD) [Roy10] est un autre modèle de mobilité conçu pour les MANET. Dans ce modèle, chaque nœud choisit aléatoirement une direction, qui est un angle entre 0 et π, et une vitesse. Par la suite, il se déplace dans la direction choisie jusqu'à atteindre le bord de la surface de simulation où il prend une pause. Une fois le temps de pause terminé, le nœud choisit aléatoirement une nouvelle direction et une nouvelle vitesse et répète le même processus. (figure 2.11)

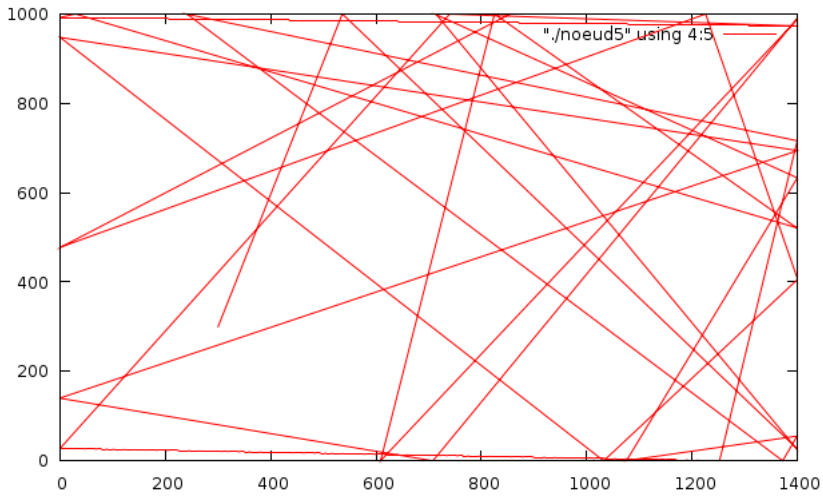


FIGURE 2.11 – Mouvements de RD

Gauss-Markov

Ce modèle peut s'adapter aux différents niveaux de hasard [Roy10] (c'est-à-dire totalement aléatoire, partiellement aléatoire ou déterministe selon la corrélation temporelle) à travers un paramètre de réglage du degré de hasard α, avec $0 \leq \alpha \leq 1$.

Initialement, une vitesse et une direction sont assignées aux différents nœuds mobiles. Pour des intervalles de temps fixes, le mouvement a lieu en mettant à jour la vitesse et la direction. Plus précisément, les valeurs de la vitesse et de la prochaine position au $n^{ème}$ instant sont calculées à partir de celles du $(n-1)^{ème}$. Pour s'assurer qu'un nœud ne reste pas trop longtemps près du bord de la zone de simulation, les nœuds sont poussés loin du bord quand ils sont à moins d'une certaine distance du bord.

Les valeurs de la vitesse, de la direction et de la position à chaque instant, dépendent des valeurs de l'instant précédent ce qui crée un mouvement plus régulier des nœuds.

La figure 2.12 présente le mouvement d'un nœud selon le modèle Gauss-Markov avec trois valeurs différentes du paramètre de réglage du degré du hasard α. Quand α est égal à 1, le mouvement est linéaire. Le nœud garde les valeurs de la vitesse et de la direction initiales. Ces valeurs ne sont modifiées que sur les bords de la zone de simulation.

Manhattan

Manhattan [Roy10], est un autre modèle de mobilité conçu pour les VANET. Il a été principalement étudié pour le mouvement dans les zones ur-

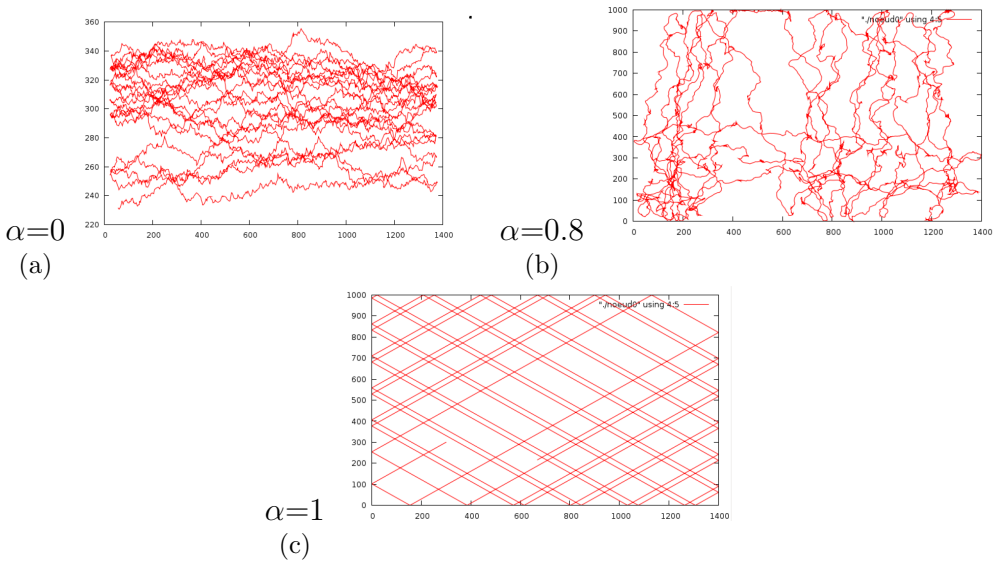


FIGURE 2.12 – Mouvements de Gauss Markov avec trois valeurs de α

baines où les rues sont organisées d'une certaine manière. Ce modèle utilise une topologie des routes représentée par une grille.

Chaque nœud mobile se déplace dans une direction horizontale ou verticale suivant la carte urbaine. A l'intersection des routes, le mobile peut tourner à gauche, à droite ou continuer tout droit. La sélection des mouvements des routes suit une approche probabiliste. En effet, la probabilité de continuer tout droit est de 0.5 tandis que tourner à gauche ou à droite est de 0.25 pour chaque direction. La vitesse du nœud mobile à un intervalle de temps dépend de la vitesse de l'intervalle de temps précédent. Elle est aussi limitée par la vitesse du nœud qui précède sur le même chemin. De plus, deux nœuds dans la même voie doivent être à une distance de sécurité SD (Safe Distance), la vitesse du nœud suivant ne pouvant pas excéder le premier.

Ce modèle de mobilité est plus adapté aux scénario des mouvements véhiculaire.

ii. Modèles de mobilité pour les UAANET

Bien que la mobilité soit l'une des caractéristiques des réseaux ad hoc de drones, il existe quelque modèles de mobilité permettant d'imiter le mouvement d'une flotte de drones.

Distributed Pheromone Repel Mobility Model

Dans [KNt06], les auteurs ont proposé deux modèles de mobilité conçus pour des missions de reconnaissance : un modèle aléatoire *Random Mobility*

Model et un autre distribué *Distributed Pheromone Repel Mobility Model*. Ces modèles de mobilité se basent sur trois actions :

- aller tout droit ;
- tourner à droite (-45^o) ;
- tourner à gauche (45^o).

Avec le *Random Mobility Model*, chaque drone choisit son mouvement en se basant sur des probabilités fixes. Alors qu'avec le *Distributed Pheromone Repel Mobility Model* une carte contenant les informations sur les mouvements des autres drones est utilisée pour guider les aéronefs. Cette carte est créée grâce à des échanges entre les drones concernant les zones scannées par les autres engins. Par conséquent, chaque nœud décide de tourner à droite, à gauche ou de continuer tout droit.

SRCM

Dans [WGWW10], Les auteurs présentent un autre modèle de mobilité conçu pour les réseaux ad hoc de drones, basé sur des mouvements semi-circulaires nommé SRCM *Semi-Random Circular Movement*. Ce modèle est idéal pour les réseaux dans lesquels les drones survolent une localisation fixe afin de collecter des informations.

Les positions initiales des nœuds leurs permettent de tourner autour d'un seul point fixe O, le centre de la zone circulaire C, chacun sur son cercle (figure 2.13).

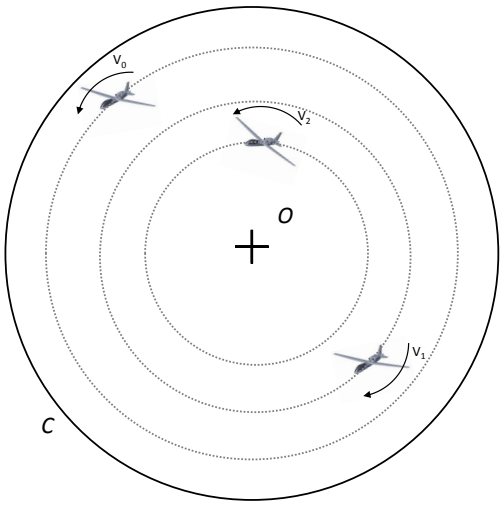


FIGURE 2.13 – Le principe de SRCM

Chaque nœud fixe une valeur de vitesse aléatoire et choisi une position sur son cercle, tout en gardant le même sens d'avancement (sens trigonométrique

ou le sens contraire). A l'arrivée à sa destination, il reste fixe pour une durée de temps fixe. Par la suite il reprend son chemin vers la position de sa nouvelle destination. Une fois que le nœud a complété le tour de son cercle, il refait la même procédure sur un autre cercle (de centre *0* et situé à l'intérieur du cercle C), choisi aléatoirement.

Modèle de mobilité pour plates-formes aériennes
Les auteurs de [XWNF13] ont proposé un autre modèle de mobilité pour les plates-formes aériennes basé aussi sur le mouvement circulaire. Chaque nœud tourne autour une position choisie aléatoirement avec une vitesse fixe jusqu'il choisit une autre position. Ce modèle est capable de changer l'altitude durant le vol en deux approches. Dans la première, l'altitude est corrélée avec les autres déplacements alors que dans la deuxième approche, les trois dimensions sont indépendantes. La corrélation entre les trois dimensions génère des mouvements avec une large variation d'altitude. Ce modèle est bien adapté à des scénarios de vol pour des applications militaires ou des spectacles. Alors que la deuxième approche, qui garde les trois dimension x, y et z indépendants, permet d'avoir moins de variation d'altitude. Ce modèle est mieux adapté aux applications civiles et commerciales.

2.3.1.2 Validation d'un nouveau modèle de mobilité

La question de la validité d'un modèle de mobilité est peu abordée dans la littérature. Certains auteurs qui ont développé de nouveaux modèles, les ont comparé aux modèles usuels comme dans [ESB+04]. Ils ont réalisé une comparaison du comportement du nouveau modèle de mobilité réaliste au comportement de plusieurs autres modèles de mobilité. Les critères utilisés dans cette étude sont la distribution spatiale des nœuds sur la zone de simulation, le taux de perte et le délai de transmission de données.

Les deux modèles de mobilité présentés dans [KNt06] ont été comparés ensemble pour étudier les performances de l'un par rapport à l'autre en utilisant comme métriques : la couverture et la connectivité des nœuds.

Dans [MM06], le nouveau modèle de mobilité a été validé en se basant sur la comparaison avec les traces de mouvement réel. Les métriques utilisées pour cette évaluation sont :

- La durée de contact : la durée de temps durant lequel deux nœuds sont à la porté l'un de l'autre ;
- Le nombre d'inter-contact : la fréquence et la probabilité d'être en contact directe avec la destination du paquet.

D'autres chercheurs préfèrent la modélisation mathématique comme dans [XWNF13] et [WGWW10].

De plus, on trouve différents métriques utilisés pour étudier les propriétés statistiques des modèles de mobilité ([YPD07],[ZHR04],[TTV07]) : Nous proposons une classification de ces métriques en deux catégories : *géométrique* et *performance réseau*. La catégorie *géométrique* ou spatiale qui permet d'étudier l'aspect physique du mouvement (les trajectoires) et son impact sur la topologie du réseau, comme :

- **Distribution spatiale des nœuds** : représente graphiquement la densité des nœuds dans la zone de simulation. Elle permet de vérifier si la distribution est homogène ou concentrée autour du centre ;
- **Degré de dépendance spatiale** : présente le degré de corrélation de vitesse de chaque nœud avec ses voisins ;
- **Degré de dépendance temporelle** : détermine le degré de variation de la vitesse et de la direction du nœud au cours du temps. Cette valeur est faible si la vitesse et la direction sont faiblement corrélées au cours du temps ;
- **Vitesse relative** : mesure la différence entre les vitesses de deux nœuds. Elle donne une indication sur le dynamique du réseau ;
- **Durée de lien** : mesure la moyenne de temps durant lequel il existe un lien radio entre deux nœuds ;
- **Taux de changement du lien** : fait référence aux changements de la topologie par le changement des états des liens entre les nœuds. En effet, si un lien entre deux hôtes est établi, on considère que l'état du lien est actif. Si le lien a été rompu à cause du mouvement du nœud, l'état du lien est considéré inactif.

La catégorie *performance du réseau* permet d'évaluer l'impact du mouvement sur les performances des protocoles de communication. Elle groupe des mesures comme le taux de perte des paquets ou le délai de transmission.

Malgré les différents facteurs de la simulation (comme par exemple les modèles de mobilité) existants, l'environnement virtuel créé par les simulateurs réseaux ne peut pas stimuler certains caractéristiques et facteurs de l'environnement réel comme par exemple, les échecs techniques qui peuvent affecter n'importe quel nœud du réseau, ou encore la puissance de réception qui reste constante dans la simulation à n'importe quel distance à l'intérieur de la portée radio de l'émetteur.

2.3.2 Expérimentation réelle

Bien que l'expérimentation réelle soit difficile à réaliser, elle reste utile et nécessaire pour évaluer les performances d'un système dans son environnement réel.

La réalisation d'une expérimentation avec des drones nécessite des matériels comme les aéronefs et la station sol ainsi que de la présence des humains pour piloter et contrôler les drones. De plus, la réalisation d'une expérimentation réelle peut être difficile à cause de plusieurs facteurs comme [BDJH04] :

- le coût : il limite le nombre de drones qu'on peut utiliser pour réaliser une expérimentation ;
- la flexibilité : elle handicape la réalisation des scénarios ainsi que leur répétition sur un grand nombre de fois ;
- surveillance du réseau : il doit avoir le minimum d'impact possible sur le déroulement de l'expérimentation. Ainsi, il est nécessaire d'utiliser le minimum de la capacité du processeur embarqué ainsi que de la bande passante disponible dans le réseau.

Malgré toutes ces difficultés l'expérimentation réelle reste une étape importante dans le processus de création des systèmes de communication.

L'article [BDJH04] présente une expérimentation réalisée pour des petits drones (10kg). Elle a pour but l'évaluation du réseau ad hoc de drones utilisant DSR comme protocole de routage et le standard 802.11b au niveau MAC.

Cette expérimentation a mis en évidence quelques défaillances de DSR, comme les problèmes de routage rencontrés avec la mobilité des nœuds notamment sa lente réaction face à l'échec des routes.

Dans [XSX+10], les auteurs ont réalisé une expérimentation afin de valider le fonctionnement d'un algorithme de formation de vol par un échange de données à travers un réseau ad hoc. Elle a montré la faisabilité, la validité et l'efficacité de l'utilisation d'un réseau ad hoc pour des fonctionnalités coopératives.

Dans le cadre du projet CARUS [CLM+11], une expérimentation a été réalisée en utilisant cinq drones. Cette réalisation avait pour but de montrer la faisabilité de l'utilisation d'une flotte de drones autonomes et coopératifs. Les échanges entres les drones sont réalisés par une diffusion asynchrone de données.

La simulation et l'expérimentation réelle peuvent être réalisées ensemble afin de bénéficier des avantages des deux. En effet, la première présente une étude facile à réaliser et reproductible malgré certaines limites d'imitation de l'environnement réel. Ces limites peuvent être rattrapées par des

expérimentations réelles permettant de valider les résultats obtenus par la simulation. Par conséquent, la simulation et l'expérimentation réelle se complètent pour une validation fiable de nouveaux protocoles de communication.

Conclusion

Les réseaux ad hoc sont caractérisés par une variété de caractéristiques comme la capacité des nœuds utilisés en termes d'autonomie, énergie, portée radio et processeur, ou encore la densité, la topologie du réseau et la mobilité. Par conséquent, ils sont classés selon leurs applications, objectifs, déploiement, ou types de communications. Les réseaux ad hoc de drones (UAANET) est l'une des classes de MANET qui présente le cas d'une flotte de drones qui coopèrent entre eux afin d'accomplir une mission. Ce genre de réseau a des spécificités par rapport aux autres classes de MANET comme la mobilité

Pour les UAANET, la communication joue un rôle très important dans le déroulement d'une opération puisque certains messages échangés peuvent être critiques et le choix des protocoles de communication utilisés peut affecter le déroulement des missions des drones. Par conséquent, un système de communication fiable doit être utilisé. Certains protocoles de communication de niveau MAC et routage ont été proposés pour être utilisés dans ce cas. La majorité est proposée pour des conditions bien spécifiques, comme les protocoles qui fonctionnent avec des types d'antenne ou de déplacement spécifiques. Certains autre protocoles nécessitent des capacités de calcul qui semble complexe par rapport à la capacité limitée des mini-drones. Par conséquent, notre choix de protocole s'est dirigé vers des protocoles qui ne nécessitent pas des conditions spécifiques et qui peuvent être implémenté sur un système Linux. Nous optons pour le standard IEEE 802.11 pour le niveau MAC et le protocole AODV pour le routage des données. L'implémentation AODV-UU est choisie puisqu'elle est disponible à la fois pour le simulateur réseau ainsi que pour le système Linux.

En outre, les UAANET ont une exigence en terme de QdS. Effectivement, au cours d'une opération, les drones échangent différents types de trafic selon la tâche de l'émetteur ou encore le protocole utilisé. Par conséquent, le système de communication utilisé pour les UAANET doit être capable de différentier le trafic et garantir différents niveau de QdS. Des mécanismes utilisés pour la gestion de la QdS dans les réseau ad hoc ont été présenté dans la deuxième section de ce chapitre. Les modèles de fourniture de QdS FQMM et SWAN permettent de différencier le trafic globalement en deux

catégories : temps-réel qui a la priorité absolue et le reste de trafic (Best-effort). Dans des situations de congestion, où les nœuds échangent trop de trafic temps-réel, le trafic Best-effort risque d'être pénalisé et dégradé ou encore d'être inexistant.

L'architecture de communication pour les UAANET doit être capable de fournir chaque trafic ses besoins en terme de QdS tout en préservant le trafic Best-effort.

Pour ce genre d'études, il est très recommandé d'évaluer les performances du système afin de prédire les problèmes possibles qui peuvent l'affecter dans des conditions réelles. La simulation et l'expérimentation réelle sont deux moyens complémentaires permettant de valider un nouveau protocole de communication. En effet, l'expérimentation réelle est plus difficile à réaliser. La simulation permet de créer un environnement virtuel permettant d'imiter la réalité mais avec des limites. Le modèle de mobilité est l'une des bases de cet environnement virtuel qui doit être très fidèle au mouvement réel des nœuds puisqu'il traduit le changement de la topologie du réseau. Par la suite, il traduit la perturbation de connexion entre les drones. Par conséquent, un modèle de mobilité reproduisant le vrai mouvement des drones est nécessaire.

Le chapitre suivant présente l'architecture de communication DAN (*DCoS Ad hoc Network*) permettant de différentier le trafic et de garantir pour chaque classe les QdS demandées dans le contexte de réseau ad hoc de drones.

Chapitre 3

Architecture DAN

Contents

3.1 Recueil d'exigences 64
3.2 Classes de trafic 67
3.3 L'architecture de DAN 68
3.3.1 API DAN 69
3.3.2 L'agent DAN 70
3.3.3 Le contrôleur d'admission 71
3.3.4 Le classificateur 71
3.4 La signalisation de DAN 72
3.5 Le Fonctionnement de DAN 73
3.5.1 API DAN 73
3.5.2 L'agent DAN 76
3.5.3 Le contrôleur d'admission 77
3.5.4 Le classificateur 78

Introduction

Ce chapitre présente une architecture de communication pour les réseaux ad hoc de drones coopératifs, nommé DAN (*DCoS Ad hoc Network*). Elle est capable de s'adapter aux différentes exigences des applications en terme de QdS et à leurs évolutions au cours du temps ainsi que de garantir une différenciation de service.

DAN combine des mécanismes de gestion de QdS indépendamment des protocoles MAC et routage pour pouvoir s'adapter aux différents systèmes multi-agents coopératifs. À partir de l'étude de l'existant présentée dans le

chapitre précédent, nous avons choisi le protocole de routage AODV et le standard IEEE 802.11 pour être utilisés dans cette architecture de communication.

L'objectif de la conception de cette architecture ne s'arrête pas au niveau de la proposition dans la littérature. L'un des objectifs de la conception de cette architecture est la réalisation et l'intégration aux drones Paparazzi réels. Pour cette raison, nous avons essayé de simplifier au maximum le fonctionnement et la combinaison de ces mécanismes pour faciliter leur implémentation réelle.

Ce chapitre présente dans la première section les exigences de conception de cette architecture de communication. À partir de l'analyse de ces exigences, des classes de services sont définies dans la deuxième section. Par la suite, la troisième section présente les différents modules qui constituent l'architecture DAN. Ensuite, la signalisation utilisée dans ce système est introduite dans la quatrième section. Finalement, le fonctionnement de toute l'architecture est détaillé dans la dernière section de ce chapitre.

3.1 Recueil d'exigences

En tant que classe des MANET, le réseau ad hoc de drones hérite de ses limites et ses problèmes. En effet, le canal de transmission sans fil est caractérisé par ses ressources limitées ainsi que sa sensibilité aux interférences et aux obstacles.
En outre, le réseau ad hoc de petits drones a d'autres spécificités qui doivent être prises en compte lors de la création et l'évaluation de nouveaux protocoles de communication pour les UAANET. Les petits drones ont une autonomie très faible, leur source d'énergie limitée ne tient au maximum qu'une vingtaine de minutes.

Au cours d'une opération réalisée par une flotte de drones, chaque aéronef peut être affecté à une tâche spécifique comme la surveillance d'une zone géographique bien déterminée ou la recherche d'une cible précise. Ces tâches peuvent nécessiter des données spécifiques en entrée permettant de diriger son évolution et envoyer différents types de messages comme résultats. Dans les opérations coopératives, l'évolution de chaque tâche peut dépendre des autres tâches obligeant les drones à échanger des données entre eux. En conséquence, les drones échangent plusieurs types de message dans toute les directions (entre eux et avec la station de contrôle).

À partir de l'étude des fichiers d'enregistrement de trafics envoyés au cours des vols réels, nous avons pu constater différents types de message que nous avons classé selon leurs contenus comme suit.

Certains messages sont échangés entre les drones et leur station de contrôle comme :

- **Informations du système (IS)** : elles sont envoyés du drone vers la station de contrôle (liaison descendante). Ces messages contiennent des informations sur l'état des aéronefs (batterie, position géographique, niveau de service, etc). Il s'agit d'un échange entre les couches hautes du système ;
- **Informations de l'opération (IO)** : elles sont envoyés également des drones vers la station sol. Ces messages peuvent avoir une variété de contenus comme la température, la localisation, la pression, ou le multimédia (son, image ou vidéo). Selon la mission, ces messages ont des priorités différentes par rapport aux autres types de trafic. Il s'agit aussi d'un échange entre les couches hautes du système ;
- **Consignes de l'opération (CO)** : elles sont envoyés de la station de contrôle vers les drones (liaison montante). Ces messages contiennent des ordres pour modifier la direction d'un aéronef, passer à une autre tâche, modifier la qualité d'image envoyée, finir l'opération et revenir, etc. Cet échange est réalisé entre les couches hautes du système ;
- **Messages de contrôle (MC)** : ils sont envoyés par les protocoles de communication (Routage, ARP, etc). Il s'agit d'un échange entre les couches basses dans les deux directions (de la station de contrôle vers les drones et inversement).

Puisque le système communique à travers un réseau ad hoc, tous ces messages peuvent passer par un ou plusieurs nœuds intermédiaires dans le but d'atteindre leur destination finale. Cependant, d'autres messages doivent être échangés entre les drones comme :

- **Information de système (IS)** : elles sont échangés selon la mission et les fonctionnalités des drones comme la localisation géographique ou la direction des voisins permettant de modifier les circonstance de la tâche. (Par exemple les échanges de positions géographiques afin de recréer le vol, allouer les tâches ou éviter les collisions) ;
- **Information de l'opération (IO)** : certaines missions nécessitent l'échange des résultats des tâches des autres drones pour pouvoir adapter leurs objectifs comme dans [SFS+04b] où les drones adaptent leurs niveaux de qualité d'image envoyés en fonction de l'importance de ses données et celles des autres qui peut varier au cours du temps selon leurs résultats de recherche ;
- **Messages de contrôle (MC)** : message de contrôle réseau (routage, ARP, MAC, ICMP, etc).

Par conséquent, la priorité de chaque type de message peut varier selon la mission, l'émetteur, le type et le contenu du message. Les demandes de chaque type de trafic en terme de QdS sont différentes, certains exigent le minimum de délai, d'autres ont besoin du maximum de débit ou de plus de fiabilité. En outre, quelques échanges nécessitent une garantie stricte sur la QdS.

Puisque les circonstances d'une opération évoluent au cours du temps, les rôles et les tâches des aéronefs peuvent changer. Par conséquent, leurs demandes en termes de QdS peut varier. Par conséquent, la classification usuelle du trafic selon le type de protocole utilisé dans certaines propositions étudiées dans le chapitre précédent n'est pas suffisante pour ce système.

La QdS, dans ce genre de système doit être gérée par un module :

1. Capable de s'adapter à la variation des tâches des drones ainsi qu'à leurs besoins en terme de QdS ;
2. Capable de respecter les exigences de chaque message lors de l'émission.

Les besoins QdS doivent être fixés par un module conscient du rôle réalisé et du degré d'importance de ses données (s'il s'agit d'une information critique ou non). Ces caractéristiques peuvent être regroupées dans les fonctionnalités du niveau applicatif. En effet, chaque application peut reconnaitre les situations critiques et les échanges qui en résultent.

Par conséquent, selon la décision prise par l'application, les autres modules de l'architecture de communication doivent respecter les exigences de chaque trafic. L'objectif est de pouvoir différencier les paquets et allouer les ressources nécessaires pour chaque classe.

L'évaluation de ce système doit être réalisée dans des conditions réalistes afin de prédire les problèmes possibles qui peuvent l'affecter dans un environnement réel. Pour notre cas, nous avons choisi deux moyens pour étudier notre système de communication. En premier temps, il sera étudié par simulation. Par la suite, dans un deuxième temps, il sera validé par des expérimentations réelles.

La simulation doit être réalisée dans un environnement proche de la réalité. En conséquence, puisque la mobilité est une particularité très importante pour les réseaux ad hoc, un modèle de mobilité spécifique reproduisant les mouvements de drones doit être utiliser. Ce modèle doit pouvoir générer une variété de cas de situations possibles.

De plus, cette architecture de communication doit être testée et utilisée en environnement réel. En conséquence, une plateforme de test doit être

créée en utilisant de vrais drones Paparazzi ainsi que leur station de contrôle permettant de les faire communiquer à travers un réseau Ad hoc conscient de leurs besoins de QdS.

3.2 Classes de trafic

Comme présenté dans la section précédente, il existe plusieurs types de messages échangés entre les drones et la station de contrôle, groupés essentiellement en quatre catégories (IS, MC, IO et CO).

Les messages de la catégorie CO sont des messages critiques permettant de modifier le déroulement de l'opération. Ils doivent être acheminés le plus tôt possible et avec le maximum de fiabilité.
Certains messages de la catégorie IS peuvent être critiques aussi, permettant d'informer la station de contrôle sur les difficultés rencontrées par les drones. Par conséquent, ces messages permettent de limiter les problèmes techniques. De plus, cette catégorie contient des messages envoyés périodiquement. La perte de certains de ces messages ne cause pas de soucis pour l'avancement de la mission. Au contraire, il est préférable de perdre ces messages que les recevoir avec un grand délai.

La catégorie de messages IO contient des messages liés à la tâche de l'aéronef qui peuvent avoir des besoins modifiables au cours du temps. Cette catégorie de trafic détermine différentes exigences de QdS. En effet, elle contient des messages envoyés périodiquement qui peuvent supporter quelques pertes ou les délais comme les messages de la classe IS. De plus, elle peut avoir des messages critiques qui demande plus de fiabilité comme la classe CO. Aussi, elle regroupe des flux de trafic temps-réel qui nécessitent un débit fixe et une garantie stricte sur la QdS.

Le trafic de la catégorie MC permet d'établir les liaisons entre les différents nœuds. Ce trafic joue un rôle important pour la continuité de l'opération. Ainsi, il ne doit pas être pénalisé par les autres flux.

Par conséquent, trois classes de trafic sont définies permettant de regrouper les différents types de messages :

Le trafic Urgent

C'est la classe des messages critiques. La réception avec délai ou la perte de ces messages peuvent causer des problèmes pour tout le système. Par conséquent, ce type de trafic nécessite le minimum de délai et le maximum de fiabilité. Essentiellement, les messages de la classe Urgent sont des messages

sporadiques, envoyés uniquement en cas de nécessité comme par exemple un résultat important, un changement de direction immédiat, une situation critique comme le niveau de l'énergie d'un drone.

Cette classe de trafic est sensible aux pertes et aux délais et elle a la priorité absolue.

Le trafic Premium

C'est la classe des flux de trafic qui ont des exigences QdS en termes de taux de perte, délai et débit. Elle nécessite une réservation de ressource pour avoir une garantie stricte sur la QdS. Le trafic temps-réel est un exemple de trafic de cette classe.

Le trafic Best-effort

Cette classe groupe le reste du trafic qui ne demande aucune garantie sur la QdS. D'ailleurs, pour les messages périodiques, il est préférable de perdre des paquets que de les recevoir avec un grand délai pour garantir que le système soit à jour. Les messages de localisation géographique et certains messages IO comme les valeurs de la température ou la pression mesurées sont des exemples du trafic Best-effort.

Chaque type d'échange est classé dans l'une de ces trois classes présentées ci-dessus. Le choix de la classe est réalisé par l'application et peut être modifié plusieurs fois au cours de l'opération selon les circonstances. Ainsi, La classification du trafic est indépendante du protocole utilisé (UDP ou TCP). Un trafic TCP peut être classé en Premium et un trafic UDP peut être un Best-effort et inversement.

La section suivante présente l'architecture de communication permettant aux applications de gérer la classification de leurs trafics et permettant de respecter les exigences de chaque classe de trafic.

3.3 L'architecture de DAN

Cette section présente les différents modules constituant l'architecture de communication DAN.

DAN (*DCoS Ad hoc Network*), est un système de communication pour les réseaux ad hoc de drones capable de différencier le service. Il groupe un

ensemble de modules et de mécanismes permettant de répondre aux exigences des UAANET.

Ces modules permettent aux applications d'exprimer les besoins de chaque flux généré à travers une API (*Application Programming Interface*) spécifique (figure 3.1) selon des critères prédéfinis comme le contenu du message. En se basant sur ces consignes, les autres modules différentient les paquets, et garantie différents niveau de QdS.

DAN différentie le trafic en trois classes : Urgent, Premium et Best-effort. La classe de trafic Urgent contient des messages critiques envoyés d'une manière sporadique avec la plus haute priorité (par exemple : message indiquant le niveau très bas d'une batterie). Ces messages exigent un délai faible et une fiabilité élevée. La classe Premium regroupe les trafics exigeants en termes de QdS. Les flux multimédia est un exemple de ces trafics. Ils nécessitent une réservation de ressources permettant de leur fournir une garantie stricte sur la QdS.

Avec DAN, les ressources sont réservées avec un protocole de signalisation qui contient des messages de réservation et de signalisation d'erreurs. Ce protocole a pour but l'établissement et la fermeture des réservations. De plus, il permet de notifier l'émetteur en cas d'erreurs. DAN se base sur les paquets de données envoyés pour rafraichir ces réservations. En effet, chaque nœud sur la route garde une idée sur l'état de la réservation (soft-state). Cette vision est mise à jour par la simple réception des paquets de données ce qui permet de limiter la consommation des ressources du réseau pour la signalisation.

Les deux classes de trafic Premium et Urgent subissent un contrôle de débit au niveau du nœud source afin d'éviter le cas de congestion totale causée par une exagération d'émission de ces deux catégories de trafic.

Tous ces mécanismes sont réalisés par des modules de fourniture de la QdS. Comme présenté dans la figure 3.1, DAN est composé d'une API spécifique permettant de connecter l'application aux autres modules qui sont : l'agent DAN, le contrôleur d'admission et le classificateur des paquets.

3.3.1 API DAN

En plus des *sockets* classiques d'UDP et TCP, utilisés pour échanger les données, l'API DAN utilise des *sockets* spécifiques permettant d'échanger des messages contenant les informations des flux.

Ces *sockets* supportent quatre types d'échanges :

– la commande de réservation (`Reservation_Command`) : c'est le message qui indique la classe du trafic à émettre. Il contient aussi les exigences

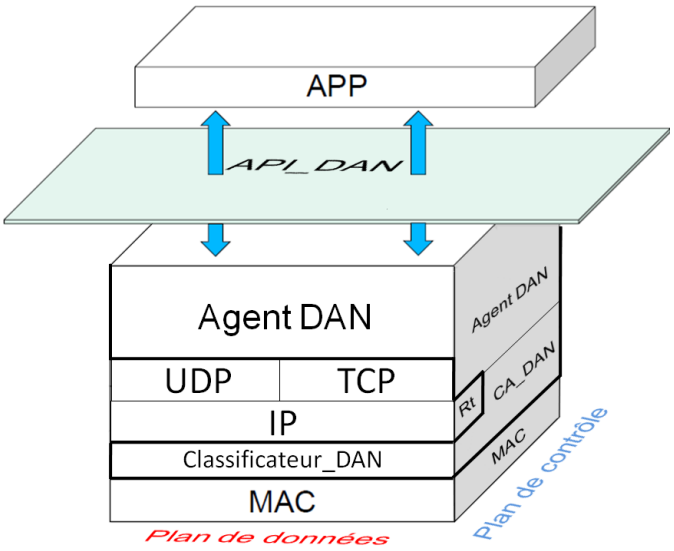


FIGURE 3.1 – Architecture de DAN

du flux Premium en terme de débit ;
- l'autorisation d'émission (`Send_Reply`) : ce message est envoyé par l'agent DAN en réponse à la demande de réservation en cas d'admission ;
- l'arrêt de l'émission (`Stop_Reply`) : si l'agent DAN n'arrive pas à réserver les ressources nécessaires pour le flux, il notifie l'application par un `Stop_Reply` contenant la valeur de débit que peut être garantie par le réseau. Par la suite, l'application a le choix d'annuler l'émission ou de négocier la réservation puisqu'elle a la possibilité de réduire ses demandes en fonction des ressources disponibles et rétablir une nouvelle réservation ;
- la commande de fermeture de réservation (`Close_Command`) : elle est envoyée par l'application à la fin de l'émission des données, ce message permet de libérer les ressources réservées pour ce flux dans le réseau.

Le fonctionnement de l'API sera détaillé dans la section 3.5.

3.3.2 L'agent DAN

L'agent DAN est responsable des autres modules constituant cette architecture. De plus, il est le contact direct de l'application. Il reçoit les demandes en termes de QdS à travers l'API. Par la suite, il définit le plan de fonctionnement des autres modules.

En cas d'insuffisance de ressources, l'agent DAN notifie l'application qui prend une décision concernant la dégradation de la qualité demandée ou l'annulation de l'émission.

En outre, ce module est responsable de l'émission et la réception des paquets de signalisation. En effet, il initialise la procédure de réservation et traite les paquets reçus en se basant sur les données fournies par les autres modules.

Une liste des flux réservés est gardée au niveau de l'agent DAN. Elle fournie les informations nécessaires aux autres modules sur les coordonnées des flux Premium admis ainsi que les messages de type Urgent.

Au niveau de chaque nœud source, l'agent DAN est responsable du contrôle de débit des deux classes de trafic Urgent et Premium afin d'éviter les cas de congestion extrême.

3.3.3 Le contrôleur d'admission

Le module de contrôleur d'admission (CA) est responsable des prises de décisions concernant l'admission de nouveaux flux. En effet, ce module est conscient des ressources localement disponibles puisqu'il garde une idée sur la capacité localement consommée (les ressources déjà allouées).

D'après les études des fichiers d'enregistrement des données des vols et en se basant sur les directions de l'équipe des développeurs du système Paparazzi de l'ENAC, nous avons constaté que le trafic Best-effort représente 63% de la totalité du trafic échangé alors que la trafic Premium ne représente que 22%. En conséquence, la capacité totale allouée au trafic Premium dans le réseau est fixée dans le but de préserver le trafic Best-effort. Elle représente la quantité maximale de bande passante réservée aux trafics Premium et elle dépend de la capacité du canal de transmission. Cette valeur est égale à 20% de la capacité totale du canal.

3.3.4 Le classificateur

En obtenant les renseignements sur les flux de l'agent DAN, le module classificateur traite les priorités entre les différents paquets qui doivent être émis en mettant en œuvre des stratégies de marquage des paquets, d'ordonnancement, de planification et de mise en file d'attente. Il retarde l'émission de certains paquets pour laisser le passage aux autres en suivant les renseignement de l'agent DAN.

Le fonctionnement de tous ces modules est détaillé dans la section 3.5.

La section suivante présente le protocole de signalisation utilisé pour réserver les ressources dans le réseau pour les flux de la classe Premium.

3.4 La signalisation de DAN

Dans le but de réserver des ressources pour le trafic Premium, DAN se base sur un échange de trois types de messages permettant d'établir des réservations, de les fermer et de notifier le nœud source en cas d'erreurs. Ces réservations sont rafraîchies par la réception des paquets data des flux concernés afin de limiter la circulation des messages de contrôle dans le réseau. Les trois types de messages échangés sont : le message de réservation, nommé *Reserve*, le message de fermeture, nommé *Close* et le message d'erreur, nommé *Error*. Ces messages sont encapsulés directement dans IP.

Le message de réservation (*Reserve*)

Le message *Reserve* est envoyé du nœud source vers la destination. La route suivie par ce message est la même qui sera sélectionnée pour le transfert des paquets data.
Le but de (*Reserve*) est d'annoncer aux nœuds intermédiaires les besoins de flux arrivant afin d'allouer les ressources nécessaires. Il contient la valeur du débit optimal permettant de garantir la qualité de trafic demandée (champ *Bw* du message) ainsi que les cordonnées du flux (adresse source, adresse destination, numéro de port source et numéro de port destination) (figure 3.2).

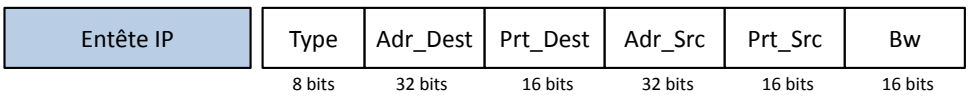


FIGURE 3.2 – Format des messages *Reserve* et *Error*

Le message d'erreur (*Error*)

Ce type de message est envoyé vers le nœud source afin de lui signaler des erreurs dans deux cas :

- un des nœuds intermédiaires ne peut pas allouer les ressources nécessaires ;
- suite à un changement de topologie, le trafic traverse une nouvelle route n'ayant pas de ressource allouée pour ce flux.

Dans les deux situations, le message d'erreur contient la valeur de débit disponible localement au niveau du nœud concerné par le problème ainsi que les coordonnées du flux (figure 3.2).

Le message de fermeture de réservation (*Close*)
Ce message a pour but d'annoncer aux nœuds intermédiaires, sur la route vers la destination, la fermeture de la réservation. Par conséquent, il permet de libérer les ressources déjà réservées. Ce message contient uniquement les coordonnées du flux concerné (figure 3.3).

FIGURE 3.3 – Format du message *Close*

Comme présenté dans les figures 3.2 et 3.3, ces trois messages sont encapsulés directement dans IP. Le rafraichissement d'une réservation est réalisé par la réception des paquets de données du flux concerné. Tant que le nœud reçoit des paquets de données ayant les mêmes coordonnées que le flux réservé, il considère que la réservation est toujours maintenue. Ce choix est pris dans le but de limiter la circulation d'autres types de message de signalisation.

La section suivante détaille les étapes suivies par DAN afin de garantir la différentiation de trafic.

3.5 Le Fonctionnement de DAN

Après avoir présenté les différents modules qui constituent l'architecture DAN ainsi que la signalisation utilisée pour réserver les ressources nécessaires, cette section détaille le fonctionnement de chaque module de DAN.

3.5.1 API DAN

Quand un nœud tente d'émettre un trafic, l'application concernée, supposée consciente des besoins de ce flux, communique avec l'agent DAN par

des échanges spécifiques à travers l'API de DAN. Le but de cet échange est de lui fournir les renseignements nécessaires sur le flux à émettre.

Chaque application demande la création d'un *socket* de l'API DAN en utilisant la commande `DAN_Socket()`. À l'aide de la commande `Bind()`, elle le lie à ses deux extrémités (l'application et l'agent DAN).

L'application initialise l'échange par la commande `Reservation_Command()`, qui contient les cordonnées du trafic à émettre ainsi que sa classe du trafic. Si ce trafic est un flux Premium, la commande doit contenir aussi la valeur de la bande passante optimale demandée.

Dans le cas d'un flux Urgent ou Best-effort, L'application commence à émettre les données immédiatement après cette commande, par contre elle doit attendre l'autorisation de l'agent dans le cas de flux Premium.

Suite à la réception de la commande de réservation, l'agent DAN initialise une procédure de réservation de ressources dans le réseau (cette procédure est détaillée dans le paragraphe 3.5.2). Selon les résultats de cette procédure, deux réponses pour l'application sont possibles :

- la réponse `Send_Reply()` qui autorise à l'application d'accomplir son émission puisque sa demande a été admise ;
- la réponse négative `Stop_Reply()`. Elle indique l'échec de la réservation et contient la quantité des ressources disponibles dans le réseau.

Dans le deuxième cas, l'application a la possibilité de négocier en réduisant ses demandes selon les ressources disponibles, si possible, dans le but de rétablir une nouvelle réservation.

A la fin de l'émission d'un flux, la commande `Close_Command()` doit être envoyée vers l'agent DAN afin de libérer les ressources réservés.

La figure 3.4 illustre un exemple de situation permettant d'expliquer le fonctionnement de l'API. Après la création du *socket* entre l'API et l'agent, l'application initialise une réservation de ressources pour un flux Premium qui a besoin de la quantité Bw1 de la bande passante.

Après l'établissement de la réservation dans le réseau, l'agent DAN répond par `Send_Reply()`. Ainsi, l'application commence sa transmission de données à travers les *sockets* UDP et TCP.

Détectant une erreur dans le réseau, l'agent notifie l'application par `Stop_Command()` contenant la quantité des ressources disponibles dans le réseau Bw2. Dans ce cas, l'application décide de négocier avec le réseau. Elle accepte de dégrader sa qualité de trafic et demande d'établir une nouvelle procédure de réservation.

Si l'application désire modifier la classe ou les demandes de son flux, elle peut notifier à tout moment l'agent DAN par la commande `Reservation_Command()`.

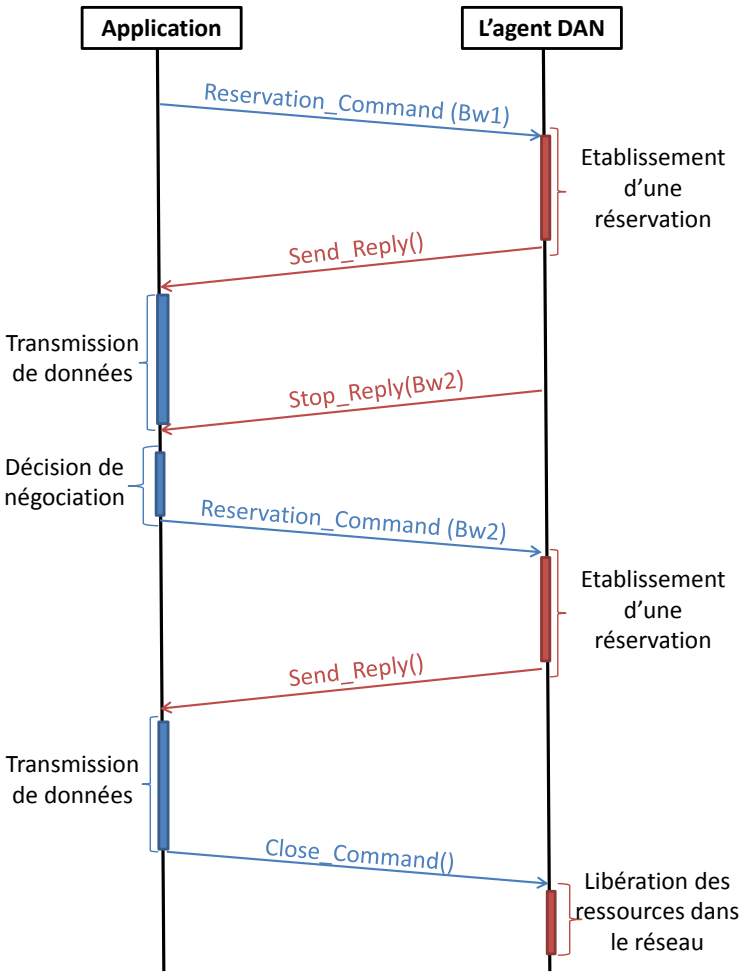


FIGURE 3.4 – Exemple de fonctionnement de l'API DAN

À la fin de la transmission, L'application envoie une `Close_Command()` à l'agent dans le but de libérer les ressources réservées.

L'API DAN permet à l'application d'être le maitre de la situation et de décider d'annuler l'émission si elle demande une QdS ferme pour son trafic ou de négocier les ressources disponibles dans le réseau. De plus, elle lui permet de modifier les besoins de chaque flux à tout moment au cours de la mission dans le but d'adapter le système à l'évolution de la tâche du nœud au cours du temps.

La sous-section suivante présente la suite de la procédure de réservation des ressources pour un flux Premium réalisé par l'agent DAN.

3.5.2 L'agent DAN

Après la réception d'une commande de réservation `Reservation_Command()`, pour un flux Premium, l'agent DAN initialise une procédure de réservation de ressources dans le réseau en envoyant le message de signalisation ***Reserve*** contenant la quantité de la bande passante optimale demandée. Ce message est envoyé en unicast vers le prochain nœud intermédiaire de la route vers la destination. Comme indiqué dans la section précédente, le message *Reserve* contient les cordonnées de la destination et du nœud source.

Après l'émission du message *Reserve*, l'agent DAN attend durant un intervalle WTS (*Wait To Send*). Si au cours de cet intervalle de temps aucun message d'erreur n'est reçu, l'agent suppose que la réservation est établie et répond à l'application par un `Send_Reply()` pour autoriser l'émission du flux.

Au niveau de chaque nœud intermédiaire, l'agent DAN consulte le contrôleur d'admission dans le but d'avoir sa décision concernant l'admission ou le refus de ce flux. Dans le cas où le nœud peut fournir les ressources nécessaires, l'agent met à jour sa liste de flux réservés et retransmet le message `Reserve` au nœud suivant. Dans le cas contraire, où le nœud ne peut pas fournir les ressources nécessaires, il rejette le message `Reserve` et envoie un message `Error` à la source contenant la quantité de la bande passante disponible localement ce qui offre une deuxième chance à la source pour négocier et réserver les ressources disponibles.

Lorsque l'agent DAN du nœud source reçoit le message d'erreur, il notifie immédiatement l'application avec le message ***Stop_Reply*** et la valeur de la bande passante disponible.

Au cours de la transmission des données, les nœuds mobiles continuent à bouger entrainant la création de nouvelles routes et le changement d'autres. Avec le changement des routes, des nœuds peuvent recevoir du trafic Premium sans réservation préalable. Dans ce cas, le nœuds traite le paquet comme du Best-effort et envoie un message d'erreur, de type ***Error***, à la source contenant la quantité de bande passante disponible dans le but de rétablir une nouvelle réservation.

Une réservation est maintenue pour une période WTC (`Wait To Clear`), et elle est rafraichie avec la réception des paquet du flux de données comme présenté dans le diagramme de séquencement 3.5. En effet, l'agent définit pour chaque flux Premium réservé un minuteur *WTC* réinitialisé avec la réception de chaque paquet de données du trafic concerné. Après une dure *WTC* de la réception du dernier paquet de données (à l'expiration du minu-

teur), le nœud considère que la réservation est fermée.

Après l'expiration de l'intervalle WTC ou à la réception d'un message `Close`, la réservation est annulée et les ressources sont libérées.

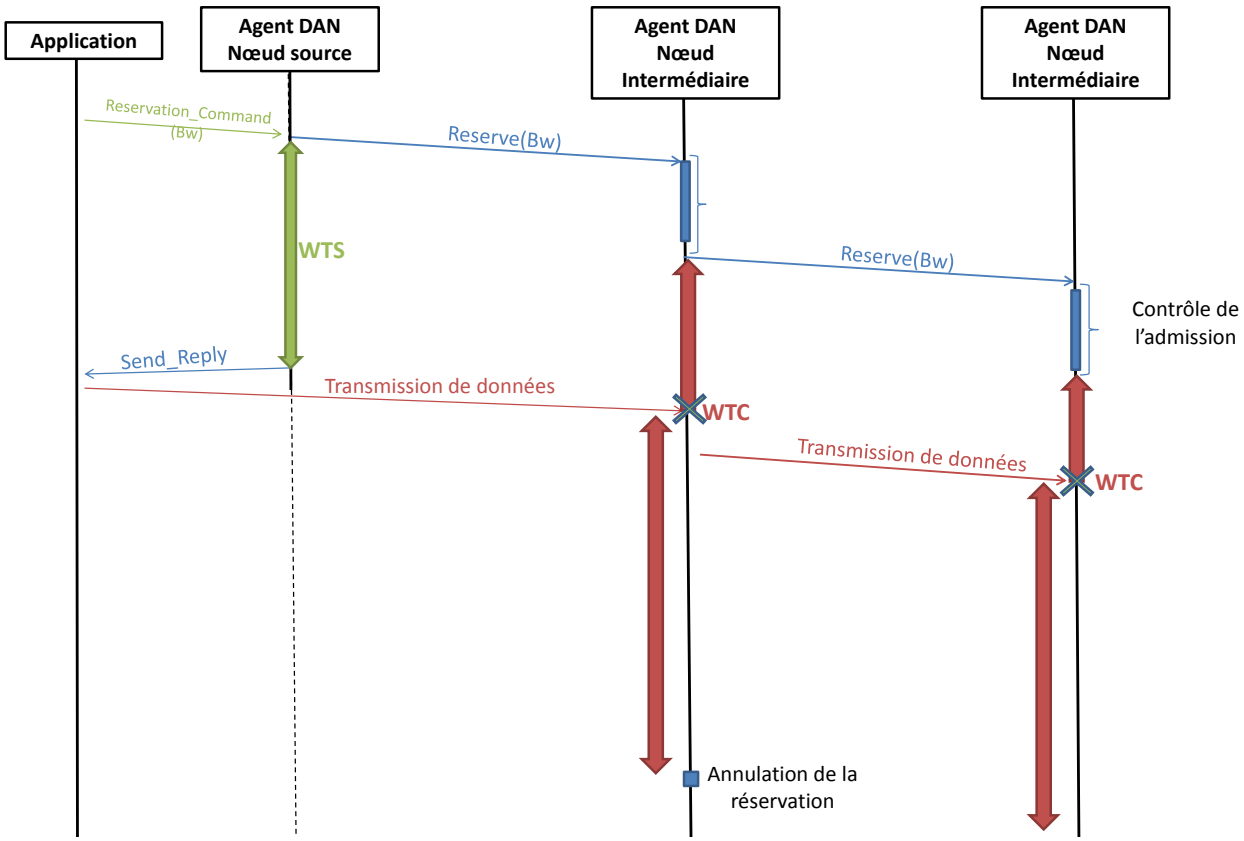


FIGURE 3.5 – Diagramme de séquencement de l'établissement et le rafraichissement des réservations

Cette sous-section a détaillé le fonctionnement de l'agent DAN qui permet de réserver les ressources pour les flux Premium. La sous-section suivante explique le fonctionnement du contrôleur d'admission.

3.5.3 Le contrôleur d'admission

Comme expliqué dans la section précédente, nous avons choisi de fixer la capacité maximale allouée aux flux Premium dans le réseau dans le but de réserver une quantité fixe pour le trafic Best-effort qui constitue la grande portion du trafic Paparazzi.

Pour garantir la QdS demandée, le contrôleur d'admission DAN garde une idée sur l'état du canal de transmission. À la réception de chaque demande de

réservation Premium, le contrôleur d'admission compare la bande passante demandée avec la quantité des ressources disponibles (la quantité de bande passante réservée pour le trafic Premium restante), si elle est plus petite, le flux sera admis sinon il sera rejeté.

Le contrôleur d'admission est le module qui prend les décisions concernant l'admission d'un flux ou non. Il permet d'allouer des ressources pour les flux Premium dans la limite de la capacité totale allouée à cette classe dans le réseau afin de garantir la QdS demandée pour chaque flux de trafic Premium.

La différenciation entre un flux Premium admis et les autres est réalisé au niveau du module classificateur présenté dans la sou-section suivante.

3.5.4 Le classificateur

En se basant sur les données de flux collectées au niveau de l'agent DAN, le classificateur différencie chaque paquet et le marque selon la classe demandée par son application. Le marquage des paquets est réalisé au niveau du champ ToS (*Type Of Service*) de l'entête IP. L'avantage de cette technique qu'elle est simple à réaliser.

Pour les paquets Premium, les nœuds intermédiaires doivent se baser, en plus du champ TOS, sur les données de l'agent DAN afin de vérifier la décision d'admission prise pour ce flux. Les paquets des flux Premium non admis sont considérés comme du trafic Best-effort. À chaque fois, que le classificateur reçoit un paquet de données d'un flux Premium admis, il informe l'agent afin de rafraichir sa réservation et réinitialiser son minuteur *WTC*.

Les paquets sont envoyés ainsi vers l'une des trois files d'attentes définies pour chaque classe de trafic (Urgent, Premium, et Best-effort). La file de la classe Urgent est la plus prioritaires alors que la file du trafic Best-effort est la moins prioritaire ce qui permet de créer la différence entre les trois catégories de trafic lors de la vidange des files.

Un ordonnanceur respectant ces priorités est utilisé (figure 3.6). À chaque fois qu'il tente de retirer un nouveau paquet pour l'émettre, il commence toujours par vérifier les files les plus prioritaire. Par conséquent, tant qu'il existe des paquets à émettre dans ces files, les autres paquets seront retardés. En cas de congestion, cette gestion des files d'attente peut aboutir à la saturation de la file la moins prioritaire (Best-effort) qui a une taille fixe. Dans ce cas, la file commence à rejeter les paquets en excès.

Les paquets de signalisation ainsi que les paquets de routage sont considérés comme du trafic Urgent dans le but de garantir le minimum de délai et de pertes pour ces échanges.

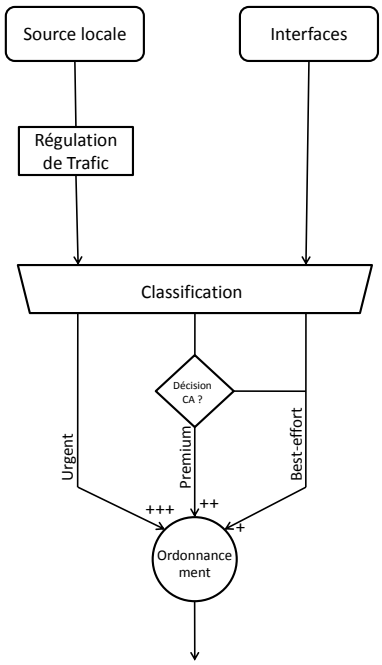


FIGURE 3.6 – Traitement de paquets par DAN

Conclusion

Ce chapitre a présenté l'architecture DAN qui répond aux exigences des systèmes de drones coopératifs. En effet, ce genre de système échange une variété de type de message qui a différents besoins de Qds. Ces besoins peuvent varier au cours du temps avec l'évolution des circonstances de l'opération des drones.

Après une étude de ces exigences et des types de messages qui peuvent être échangés au cours d'une opération d'une flotte de drones, nous avons définie trois catégorie de trafic : une classe pour le trafic Urgent qui contient les messages critiques, une classe Premium pour les flux qui ont des demandes strictes en termes de QdS et une classe pour le trafic Best-effort qui n'a aucune exigence sur la QdS.

L'architecture DAN regroupe des mécanismes de fourniture de QdS choisis pour pouvoir s'adapter à ce genre de système et aux autres protocoles MAC et routage ainsi que pour pouvoir l'implémenter facilement sur de vrais drones Paparazzi puisqu'elle ne nécessite pas des modifications complexes (comme pour la modification de l'entête IP proposé par INSIGNIA).

DAN est adapté à l'environnement et aux trafics Paparazzi puisqu'il différencie le trafic en trois classes différentes, contrairement à la majorité des modèles proposés dans la littérature qui donne la priorité absolue aux trafics temps-réel quitte à bloquer totalement les échanges Best-effort. DAN offre la haute priorité à la classe Urgent, cependant, la définition de cette classe ne la permet pas de saturer le canal puisqu'elle contient des messages échangés sporadiquement. De plus, DAN garantit la QdS pour les flux Premium par des réservations de ressources sans pénaliser et dégrader le trafic Best-effort. En effet, nous avons fixé la capacité totale allouée pour le trafic Premium dans le réseau. Cette valeur dépasse la quantité de flux Premium du trafic Paparazzi et elle sert à éviter les situations d'exagération afin de préserver le trafic Best-effort.
DAN se base sur les deux type de granularité *par flux* et *par classe* pour assurer les besoins des flux Premium ainsi que des autres classes de trafic Urgent et Best-effort. Ce principe permet de limiter les échanges de signalisation dans le réseau puisque DAN ne réserve des ressources que pour les flux Premium ce qui permet de conserver la bande passante dans le réseau.

En outre, grâce à son API, DAN permet à chaque application de négocier la qualité de son trafic et même de modifier les besoins à tout moment. Cela différencie DAN des autres structures de gestion de QdS classiques qui se basent sur le type de protocole pour différencier le trafic tout au long de l'échange.

Chapitre 4

Évaluation des performances de DAN par simulation

Contents

4.1 Modèle de mobilité PPRZM 82
4.1.1 Les mouvements des drones Paparazzi 82
4.1.2 PPRZM 85
4.1.3 Validation du modèle PPRZM 88
4.2 Évaluation de DAN 92
4.2.1 Évaluation du fonctionnement du Contrôleur d'admission 93
4.2.2 Étude des minuteurs 97
4.2.3 Performances générales de DAN 106
4.2.4 Évaluation de l'impact de variation de la charge du réseau 109
4.2.5 Évaluation avec d' autres protocoles de routage . . 112

Introduction

Ce chapitre présente l'étude de l'évaluation des performances réalisée pour l'architecture DAN par simulation.

La simulation est un moyen d'évaluation des performances de systèmes de communication et de prédire les problèmes possibles qui peuvent l'affecter en environnement réel. DAN a été étudiée en utilisant le simulateur OMNET++.

Afin de créer un environnement de simulation proche de la réalité, nous avons développé un modèle de mobilité reproduisant les mouvements réels des

drones Paparazzi. Ce modèle est présenté dans la première section de ce chapitre. La deuxième section de ce chapitre présente les différents scénarios simulés dans le but d'évaluer les performances de DAN ainsi que leurs résultats.

4.1 Modèle de mobilité PPRZM

Étant donné que les traces réelles n'offrent pas la possibilité de faire varier des paramètres de simulations dont par exemple la durée ou le nombre de drones utilisés à cause de la complexité d'obtention une variété de ces données, il est plus judicieux d'utiliser un modèle de mobilité qui reproduit le mouvement réel des drones Paparazzi dans le but d'étudier divers cas possibles.

4.1.1 Les mouvements des drones Paparazzi

D'après l'équipe de développement du système Paparazzi de l'ENAC, ainsi que l'étude des fichiers d'enregistrement des données de vols réels, les drones Paparazzi ont cinq types de mouvements possibles, nous l'avons nommé comme suit :

Mouvement *rectiligne*

Il s'agit du mouvement rectiligne vers une position de destination. Le drone a la possibilité de faire des allers-retours rectilignes en effectuant un tour complet pour faire demi-tour à chaque extrémité. Ce mouvement est défini à partir de deux positions qui sont la position initiale et la position destination (figure 4.1).

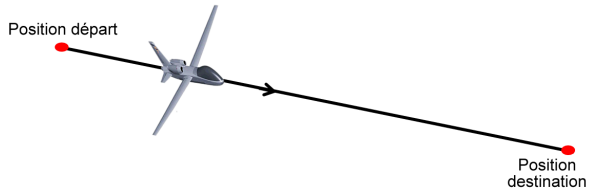


FIGURE 4.1 – Le mouvement *rectiligne*

Mouvement *circulaire*

Le drone survole une position fixe suivant une trajectoire circulaire (figure 4.2). Les mouvements circulaires sont définis par la position du centre et le rayon de la trajectoire. Pour les drones Paparazzi, le rayon *Rc* est une valeur fixe pour tous les mouvements *circulaire*, alors que la position du centre est une variable choisie pour chaque mouvement permettant de localiser ce mouvement dans la zone.

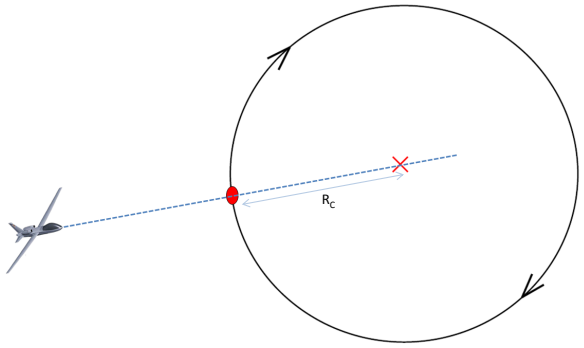


FIGURE 4.2 – Le mouvement *circulaire*

Le drone initie son mouvement par le point du cercle le plus proche de sa position actuelle. Ce point est l'intersection entre le cercle et la droite qui passe par son centre et la position du drone (figure 4.2).

Mouvement *oblong*

Ce mouvement est formé par des allers-retours rectilignes décalés entre deux points fixes avec un demi-tour une fois passé chaque point (figure 4.3). Ces deux points permettent de situer le mouvement dans la zone. Les rayons des deux demi-cercles ont la même valeur que le rayon *Rc* du mouvement *circulaire*.

Le drone initie son mouvement par l'intersection entre le premier demi-cercle de centre C_1 et le mouvement rectiligne (Point rouge sur la figure 4.3).

Mouvement *Huit*

Le mouvement *Huit* a le même principe que le mouvement *oblong*. La seule différence est que les allers-retours du mouvement *Huit* se croisent (figure 4.4) ;

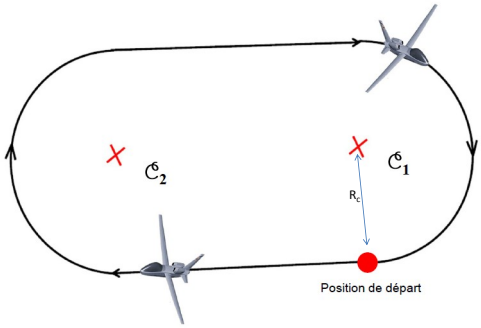


FIGURE 4.3 – Le mouvement *oblong*

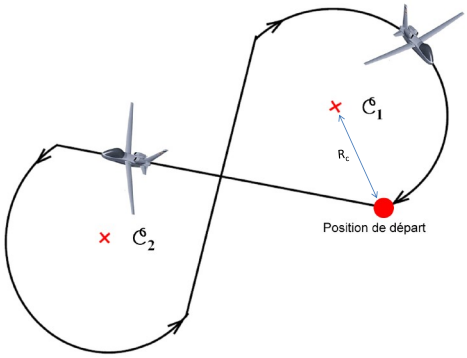


FIGURE 4.4 – Le mouvement *Huit*

Comme le mouvement *oblong*, le *Huit* se produit autour de deux points localisant la trajectoire avec la même valeur de rayon que les mouvements *circulaire* et *oblong*. Le drone initie son mouvement par l'intersection entre le premier demi-cercle de centre C_1 et le mouvement rectiligne (Point rouge sur la figure 4.4).

Mouvement *balayage*

Ce mouvement réalise des allers-retours décalés d'une manière continue permettant de couvrir la surface d'un rectangle désigné par deux de ses coins opposés (figure 4.5). Les déviations du drone sont réalisées par des demi-cercles de rayon *Rs* différents du rayon *Rc*.

La position de départ est l'un des deux points définissant la localisation du mouvement. Une fois arrivé à l'autre coin du rectangle à balayer, le drone reprend son chemin en le décalant par un demi-rayon *Rs*.

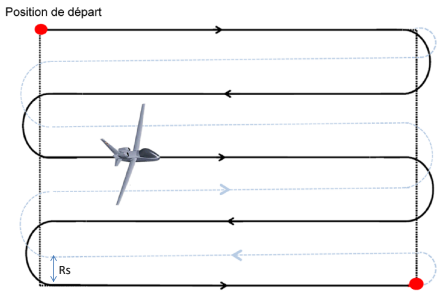


FIGURE 4.5 – Le mouvement *balayage*

4.1.2 PPRZM

Le modèle de mobilité PPRZM (*PaPaRaZzi Mobility*) reproduit ces mouvements au niveau du simulateur réseau afin de créer un déplacement de nœuds réaliste. Il a été défini selon les indications de l'équipe développeur du système Paparazzi à l'ENAC ainsi que l'étude des traces de mouvements réels.

Au début de chaque simulation, tous les nœuds se concentrent dans la même position initiale qui représente le point de décollage des drones. Par la suite, ils initialisent leurs mouvements par une phase de décollage pour atteindre les altitudes choisies aléatoirement pour chaque drone (maximum 8 m). Chaque aéronef garde son altitude fixe durant toute la durée de simulation. Les figures 4.6 et 4.7 présentent graphiquement des traces de mouvements réels de trois drones Paparazzi.

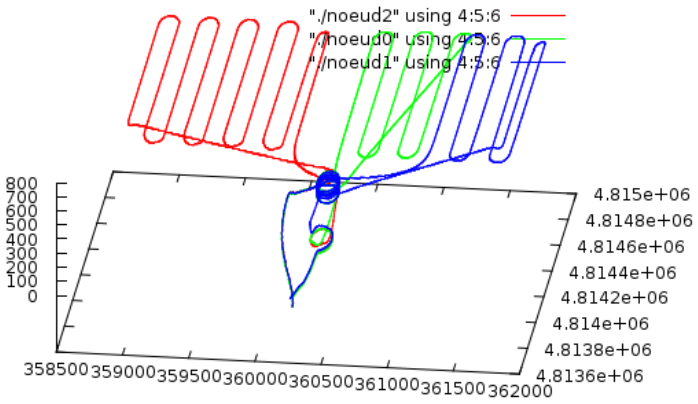


FIGURE 4.6 – Traces réelles de trois drones Paparazzi

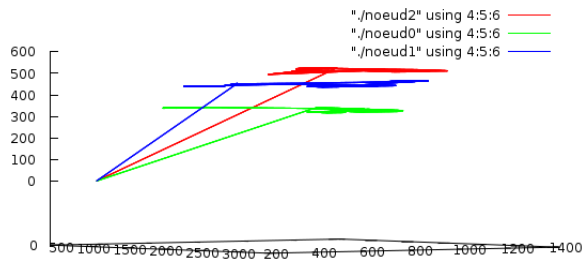


FIGURE 4.7 – Altitudes des trois drones Paparazzi utilisés dans le graphe 4.6

Au cours de la simulation, chaque drone choisit aléatoirement un type de mouvement. Par la suite il fixe ses deux caractéristiques :

- la localisation : selon le type de mouvement choisi, le drone fixe la zone d'action définie par les positions centrales dans le cas des mouvements *circulaire*, *oblong* et *Huit* ou les positions initiales et finales dans le cas des mouvements *rectiligne* et *balayage* ;
- La vitesse : c'est une valeur aléatoire uniforme choisie entre 15 m/s et 25 m/s.

Les rayons des mouvements *circulaire*, *oblong* et *Huit* ont une valeur commune différente de celle du rayon du mouvement *balayage*. Ces valeurs sont définies au début de la simulation comme un paramètre d'entrée (pour le cas réel des drones Paparazzi, le rayon des *circulaire*, *oblong*, et *Huit* est de 80 m alors que le rayon du mouvement *balayage* est de 75m).

Après avoir choisi aléatoirement un type de mouvement ainsi que sa localisation dans la zone de simulation, le drone calcule la position optimale pour initialiser le mouvement. Il se dirige vers cette position par un mouvement *rectiligne*. Une fois arrivé, le drone commence son mouvement choisi (figure 4.8). Ces différentes étapes sont reproduites périodiquement durant toute la durée de la simulation.

Les différents types de mouvements ont une probabilité plus ou moins grande de se reproduire. Par exemple, si les mouvements *circulaire*, *oblong* et *balayage* se produisent chacun 30% du temps, *Huit* et *rectiligne* se produisent chacun uniquement 5% du temps. Le choix du type de mouvement n'a aucune relation avec le choix précédent.

Ces probabilités sont paramétrables au début de chaque simulation. Cependant, d'après les experts Paparazzi et l'observation des traces de mouvements réels, pour le cas général des drones Paparazzi, les mouvements *circulaire*, *oblong* et *Scan* sont les mouvements les plus produits. Ce modèle

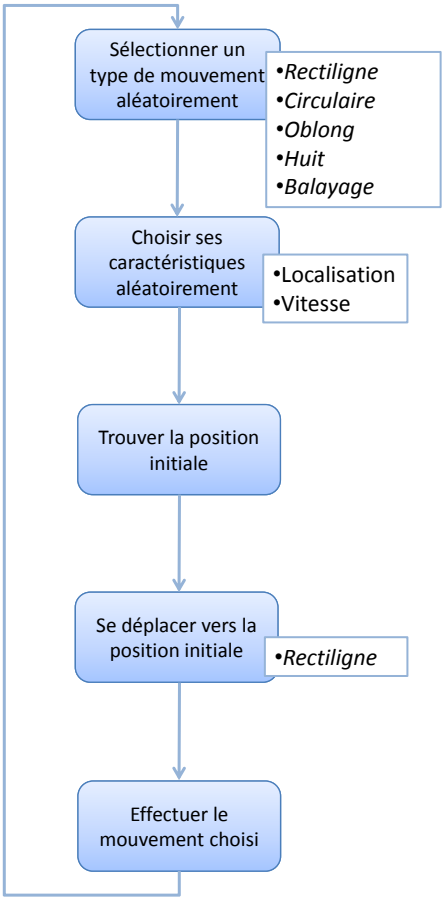


FIGURE 4.8 – Fonctionnement de PPRZM

de mobilité peut être utilisé pour l'évaluation des systèmes de communication dans le contexte d'une flotte de drones coopératifs.
Les valeurs des probabilités allouées à chaque type de mouvement sont modifiables (comme un paramètre d'entrée) dans le but d'offrir plus de degré de liberté à l'utilisateur pour adapter le modèle à son application. Par exemple, pour étudier les performances du système dans le cadre d'une mission de collecte d'information autour d'une position fixe, il suffit de fixer la probabilité du mouvement *circulaire* égal à 1 et celles de tous les autres mouvements à 0.

4.1.3 Validation du modèle PPRZM

Comme présenté dans le deuxième chapitre, il n'existe pas une méthode commune permettant de valider et évaluer un modèle de mobilité. De plus, une grande variété de critères est utilisée pour évaluer les nouveaux modèles. En conséquence nous avons fixé une méthode permettant de comparer le comportement de PPRZM avec les traces de mouvement de drones Paparazzi réels.

4.1.3.1 Méthode de validation

Pour étudier le comportement de PPRZM, nous avons classé les métriques utilisées en deux catégories : *géométrique* et *performances réseau*.

La catégorie géométrique ou spatiale permet d'étudier l'aspect physique du mouvement (les trajectoires) et son impact sur la topologie du réseau alors que la catégorie liée aux performances du réseau permet d'évaluer l'influence du mouvement sur le comportement des protocoles de communication.

Pour notre étude, nous avons choisi de comparer le comportement de PPRZM avec le comportement du modèle RWP (*Random Way-Point*) ainsi qu'avec des traces de mouvements réels de drones Paparazzi.

Pour cela nous avons sélectionné cinq mesures géographiques et cinq mesures de performances réseau pour réaliser une comparaison complète.

Métriques géographiques

Cinq mesures ont été sélectionnées :

- **La moyenne des cases vides (*EmptyCells*)** : cette métrique consiste à diviser l'espace en cases de même taille. Le nombre de cases est égal au nombre de mobiles. Par conséquent, les nœuds sont distribués uniformément, si en moyenne un drone est localisé par case. Cette mesure calcule la moyenne du nombre des cases vides au cours du temps. Elle se réfère à l'uniformité de distribution des drones dans l'espace de simulation ;
- **La fréquence de passage par une case (*Frequency*)** : Cette métrique se base sur le même principe de division de la zone étudiée en des cases. Elle compte le nombre de fois qu'un drone traverse une case. Elle calcule ensuite la moyenne sur le nombre des cases visités puis la moyenne sur l'ensemble des drones. La valeur de *Frequency* est très importante si chaque drone ne visite qu'un nombre restreint de cases (figure 4.9). Cette métrique donnent une idée sur l'uniformité de mouvement de chaque drone.

0	0	0	0
0	3	3	0
0	4	4	0
0	0	0	0

FIGURE 4.9 – Exemple de calcul de métrique *Frequency*

La figure 4.9 présente un exemple de zone de simulation divisée en grille, chaque cellule contient la fréquence de passage d'un drone par cette case. *Frequency* de chaque cellule est le rapport de la fréquence par le nombre des cellules visitées (soit 4 dans ce cas) ;

- **Le nombre de voisins(*Neighbor*)** : c'est la moyenne du nombre des voisins par aéronef. Deux nœuds sont dits voisins si l'un se trouve à l'intérieur de la portée de communication de l'autre. Cette métrique souligne la possibilité de création d'un réseau multi-saut. Une valeur importante du nombre de voisins peut affecter les performances du réseau puisque ça renforce l'interférence ;
- **Les rencontres (*Meeting*)** : elle compte le nombre de liens créés et rompus entre les nœuds au cours de la simulation. Cette métrique vise à cerner la stabilité du réseau. Plus cette valeur est importante, moins le réseau est stable ;
- **Regroupement des nœuds (*Clustering*)** : elle permet de savoir si un modèle a tendance à regrouper les nœuds dans des zones séparées. Elle compte le nombre de voisins d'un mobile et le compare au nombre des voisins de ses voisins. Plus le rapport est important plus cette mesure est élevée, les nœuds ayant tendance à s'agglutiner.

Ces métriques révèlent les caractéristiques spatiales de la topologie du réseau. Elles donnent une intuition de la difficulté que peut avoir le protocole de communication utilisé (comme la possibilité ou non de créer un réseau multi-saut, ou la possibilité de créer des groupes déconnectés au sein du même réseau, etc). Cependant, elles ne montrent pas l'influence de ce comportement sur les performances du système. Pour cette raison, nous avons choisi d'autres métriques capables d'étudier les performances du réseau.

Métriques des performances réseau

Cinq métriques permettant d'étudier les performances du réseau ont été sélectionnées :

- **Taux de livraison (*Delivery Ratio*)** : c'est la comparaison entre le nombre de paquets reçus et le nombre de paquets émis au total. Cette métrique calcule la moyenne sur le nombre de drones ;
- **Le délai de bout en bout (*End To End Delay*)** : elle calcule la valeur moyenne des écarts entre l'émission et la réception de chaque paquet de données échangé dans le réseau ;
- **Nombre de saut (*HopCount*)** : c'est la valeur moyenne des nombres de nœuds intermédiaires utilisés pour acheminer chaque paquet à sa destination ;
- **Débit dans le réseau (*Rate*)** : elle mesure le nombre de bits émis par intervalle de temps ;
- **Le nombre de paquet *RREQ*** : cette métrique est spécifique au protocole de routage AODV utilisé. Elle consiste à calculer le nombre de paquets de routage RREQ envoyés dans le but de donner une estimation sur la difficulté d'établissement d'une route dans le réseau.

Ces métriques peuvent avoir différents poids qui sont définis selon les besoins de l'application du modèle de mobilité. Pour notre étude, nous visons à valider le modèle dans le cas général. Pour cette raison, nous posons que toutes les mesures sélectionnées aient le même poids.

4.1.3.2 Résultats de la validation

PPRZM est implémenté en C++ sous OMNET++. Une trajectoire d'un nœud selon le modèle implémenté est présentée dans la figure 4.10.

Ce modèle est validé par une comparaison de comportement avec les traces réelles et le modèle de mobilité le plus utilisé RWP (*Random Way-Point*). Le même scénario, présenté dans le tableau 4.1, est utilisé avec les trois modèles de mobilité.

Comme avec les traces réelles, les drones restent immobiles dans leurs positions initiales durant les premières 10 secondes. Par la suite, ils commencent à bouger jusqu'à la fin de la simulation. Durant toute la simulation, les drones échangent des messages UDP de 64 octets à chaque seconde à travers un réseau ad hoc utilisant le protocole de routage AODV ce qui nous a permis de mesurer les performances du réseau en termes de délai de bout en bout, taux de livraison, etc.

Les résultats sont présentés dans des kiviats dans le but de faciliter la comparaison entre les différents modèles utilisés. Ces diagrammes présentent

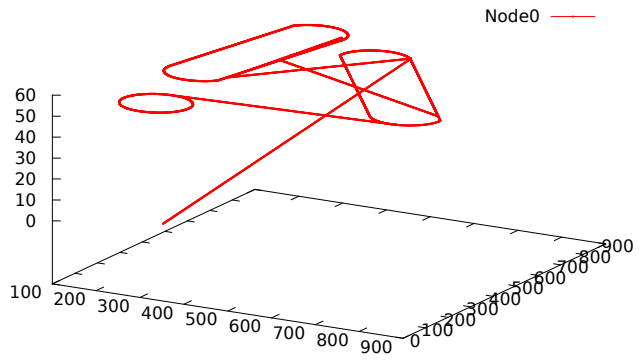


FIGURE 4.10 – Trajectoire suivie selon PPRZM

les résultats en fonction de la valeur maximale obtenue pour chaque métrique (parmi les résultats des trois modèles de mobilité).

La figure 4.11 présente les résultats géométriques des trois modèles PPRZM, RWP ainsi que les traces réelles des drones Paparazzi.

Au premier coup d'œil au graphique, nous observons les différents comportements des trois modèles simulés ce qui confirme que chaque modèle a un impact unique sur le système.

Les mesures *EmptyCells*, *Frequency* et *Meeting* du modèle PPRZM sont les plus proches de celles des traces réelles (Par exemple, *EmptyCells* du PPRZM est égale à 93% de la valeur obtenue avec les traces réelles contre 85% pour RWP). Globalement, il est possible de dire que PPRZM génère un comportement plus proche de celui des mouvements réels des drones Paparazzi puisque il a trois résultats plus proches à ceux des traces réelles que le modèle RWP.

La figure 4.12, présente les mesures des performances du réseau pour les trois modèles PPRZM, RWP et les traces réelles. Le diagramme de kiviat, montre aussi la différence du comportement entre les trois modèles de mobilité. PPRZM réalise trois meilleurs résultats que RWP sur cinq : *End To End Delay*, *Rreq number* et le *Hop Count* (Pour *RREQ Number*, PPRZM réalise 95% du nombre de Rreq envoyé avec les traces réelles, alors que RWP échange 112%).

Globalement, PPRZM a un comportement proche de celui des mouvements réels des drones Paparazzi.

Nombre de drones		40
Durée de la simulation		1000 sec
Zone de la simulation		1500 m x 1500 m
Protocole de routage évalué		AODV
Protocole MAC		802.11
Capacité du canal		54 Mbps
Trafic par drone		UDP (64 Octets/s)
Portée de transmission		100 m
PPRZM	Vitesse	valeur aléatoire uniforme entre [15 mps, 25 mps]
	Rayon du mouvement *balayage*	75 m
	Rayon des autres mouvements circulaires	80 m

TABLE 4.1 – Scénario de la simulation

PPRZM met en œuvre certains types de mouvement qui ne sont pas pris en compte par RWP comme le mouvement circulaire autour d'une position fixe (*circulaire*) ou le fait de refaire le même mouvement dans la même zone durant un intervalle de temps.

De plus, PPRZM est un modèle stochastique permettant de varier les paramètres de simulation et de générer une diversité de cas possibles ce qui est difficile à réaliser avec les traces réelles.

4.2 Évaluation de DAN

Cette section présente l'étude de DAN par simulation. Cette étude a pour but l'évaluation de DAN ainsi que la validation du fonctionnement de ses mécanismes dans différentes situations. Pour cela, cinq scénarios ont été définis :

- **1er scénario** : une validation du fonctionnement du contrôleur d'admission et la capacité du système à respecter les décisions concernant l'admission d'un flux Premium ;
- **2ème scénario** : une étude permettant de borner les durées des minuteurs *WTC* et *WTS* ;
- **3ème scénario** : une étude des performances de DAN dans un réseau congestionné ;
- **4ème scénario** : une évaluation du comportement de DAN en variant la charge dans le réseau ;

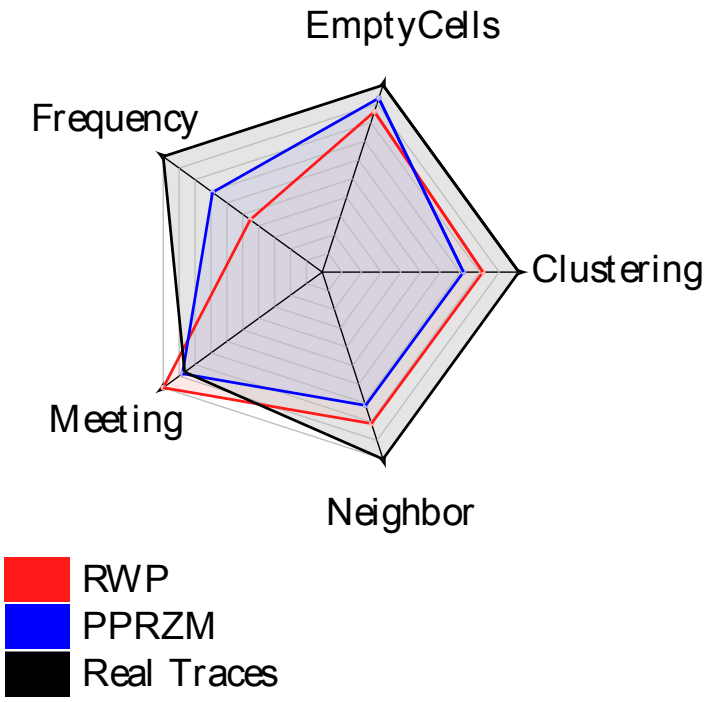


FIGURE 4.11 – Comparaison géométrique

– **5ème scénario** : une validation du fonctionnement de DAN avec d'autres protocoles de routage.

Le trafic échangé est en UDP afin de pouvoir mesurer les pertes. Toutes ces simulations sont lancées plusieurs fois dans le but de générer une variété de cas possibles. Les résultats présentés dans cette section sont la moyenne des résultats de toutes ces répétitions.

4.2.1 Évaluation du fonctionnement du Contrôleur d'admission

Le but de cette étude est de valider le fonctionnement du contrôleur d'admission de DAN. Plus précisément, elle étudie la capacité de DAN à prendre des décisions d'admission (accepter ou refuser une demande de réservation de ressources) ainsi que sa capacité à respecter cette décision et à fournir la QdS demandée à chaque flux Premium admis. Pour cette raison, nous avons fixé un scénario permettant de réaliser cette étude. Un nœud émetteur envoie vers un nœud récepteur trois flux Premium avec des demandes en termes de QdS supérieures à la capacité totale du système (Tableau 4.2).

Uniquement deux flux peuvent être admis par le contrôleur d'admission

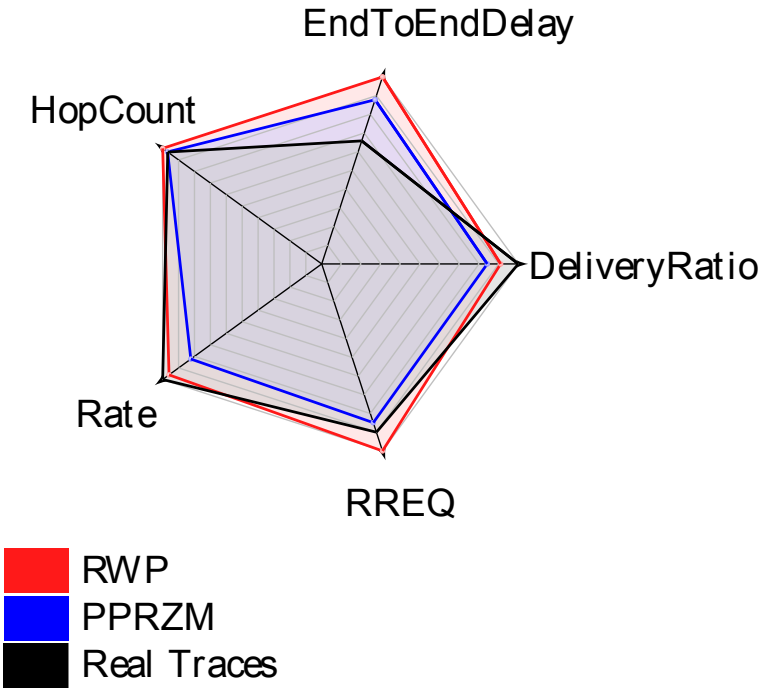


FIGURE 4.12 – Comparaison des performances du réseau

(soit le 1er et le 2eme, soit le 2eme et le 3eme). L'admission est réalisée selon l'ordre chronologique de réception des requêtes de réservation.

L'application du flux non admis par le contrôleur d'accès, sera informée par l'incapacité du système à garantir la QdS demandée. Pour cette étude, nous considérons que l'application continuera à émettre du trafic avec la qualité demandée durant toute la simulation bien qu'elle soit consciente de l'insuffisance des ressources. Par conséquent, le système va devoir faire face à un trafic Premium non admis.

Cette hypothèse permet d'étudier le comportement de DAN à la réception du trafic Premium qui n'a pas été admis par le contrôleur d'admission. Cette section présente les résultats de cette simulation en termes de taux de perte, de délais de bout en bout et de débit de transmission.

Taux de perte

Différents facteurs peuvent causer des pertes de paquets durant l'acheminement d'un trafic de la source vers sa destination comme la congestion du réseau, l'interférence et les collisions entre les paquets. Pour bien étudier le fonctionnement des mécanismes de DAN nous nous intéressons aux pertes dans le système en général en nous focalisant sur les pertes dues à la congestion des files d'attentes.

Nombre de nœuds	2
Capacité du canal radio	54 Mbps
Capacité maximale du Premium (dans le CA)	20 Mbps
Flux Premium 1	10 Mbps : 1250 Octets chaque 1 ms
Flux Premium 2	5 Mbps : 1250 Octets chaque 2 ms
Flux Premium 3	15 Mbps : 1125 Octets chaque 0,6 ms
Durée de la simulation	650 sec

TABLE 4.2 – Premier scénario

	Flux 1	Flux 2	Flux 3
Pertes dues à la congestion	0	0	55.05
Pertes totales	1.7894	1.7895	56.201

TABLE 4.3 – Taux de perte mesuré pour chaque Flux (%)

Le tableau 4.3 présente le taux de perte mesuré pour chaque flux de trafic.

Nous remarquons une grande différence entre les taux de pertes généraux mesurés pour le flux 3 (56.2%) et les taux de perte des flux 1 et 2 (1,78% pour chacun). Cet écart est dû à la différence de traitement des trois flux par les files d'attentes. En effet, la congestion du système n'a causé aucune perte de paquets des flux 1 et 2 bien qu'elle a causé la destruction de 55% des paquets du flux 3. Il est évident que les trois flux n'ont pas été traités par la même file d'attente bien qu'ils sont les trois de la classe Premium.

Cela s'explique par le fait que le système ne considère que les flux 1 et 2 comme Premium et il considère le flux 3 comme un trafic Best-effort puisque il s'agit d'un flux Premium non admis. Par conséquent, le système retarde l'émission de ces paquets dans le but de garantir la priorité des autres flux, ce qui congestionne le système et cause les pertes.

Délai de bout en bout

Le délai de bout en bout est la durée de transfert d'un paquet entre l'émetteur et son récepteur final. Plus précisément, c'est la somme de la durée de propagation du paquet le long du chemin vers la destination et de la durée de traitement réalisé par le système (comme la durée d'attente dans les files).

Le tableau 4.4 présente les délais de bout en bout mesurés pour les différents flux. La différence entre les trois flux est très remarquable (134

	Flux1	Flux2	Flux3
Délai de bout en bout (ms)	1.61	1.45	134.16

TABLE 4.4 – Délai de bout en bout mesuré pour chaque flux (ms)

ms pour le flux 3 et environ 1.5 ms pour les flux 1 et 2). Cette différence montre que le système retarde les paquets du flux 3 pour garantir la priorité demandée aux autres flux.

	Flow1	Flow2	Flow3
Temps d'attente (ms)	0.27	0.3	133.1

TABLE 4.5 – Temps d'attente dans la file d'attente pour chaque flux (ms)

Le tableau 4.5 confirme cette hypothèse. Il représente le temps d'attente des paquets de chaque flux dans leurs files d'attente. Les paquets du flux 3 doivent attendre en moyenne 133 ms avant d'être émis alors que les paquets des flux 1 et 2 n'attendent que 0,3 ms. Ces résultats confirment la conclusion que DAN garantit la QdS demandée aux flux Premium admis par le contrôleur d'admission.

Débit de transmission

Les valeurs présentées sur la courbe 4.13 sont les moyennes, sur des intervalles de 10 sec, du débit mesuré chaque 0,1 sec au niveau du récepteur.

La figure 4.13 ainsi que le tableau 4.6 présentent les débits mesurés pour les flux 1, 2 et 3.

	Min	Max	Moyenne
Flux 1 et 2	14999,7	15000,61,23	15000
Flux 3	7471	7510,8	7494,577

TABLE 4.6 – Débits mesurés pour chaque flux de trafic

Les résultats montrent que le débit de chaque flux 1 et 2 a été respecté (15 Mbits/s pour les deux) alors que le débit du flux 3 a été dégradé de moitié.

Les résultats de cette étude montrent une différence de traitement entre les trois flux de trafic Premium reçus. En effet, ayant reçu les demandes de réservation de ressources des flux 1 et 2 en premier, le CA accepte l'admission de ces flux et réserve les ressources nécessaires. En conséquence, le système

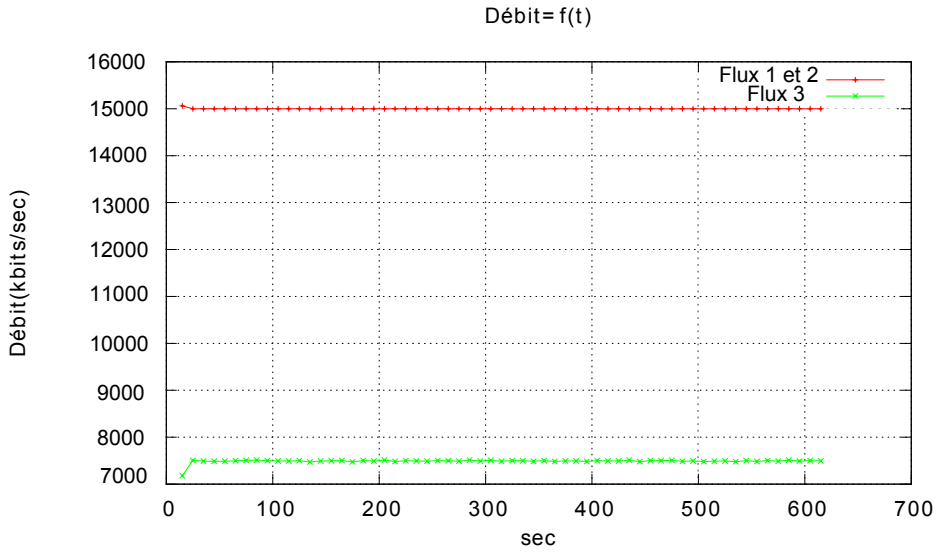


FIGURE 4.13 – Débits mesurés durant la simulation pour chaque flux de trafic

n'a plus la capacité nécessaire pour l'admission de troisième flux. Pour cette raison, il a traité ses paquets comme Best-effort puisque nous avons admis pour ce scénario que l'application continue à émettre son trafic même en absence de ressources nécessaires. Le flux 3 a été retardé dans le but de garantir la priorité demandée aux flux admis. Ainsi, il a le taux de perte et le délai de bout en bout les plus élevés. De plus, son débit a été dégradé afin de garantir les ressources nécessaires pour les flux 1 et 2.

Cette étude montre que DAN est capable d'accepter et de refuser les demandes de réservation de ressources pour les trafics Premium dans les limites de sa capacité. En outre, il offre aux flux Premium admis par le contrôleur d'admission leurs besoins en termes de pertes, de délais de bout en bout et de débit de transmission.

4.2.2 Étude des minuteurs

Cette étude a pour but de borner la durée des deux minuteurs *WTS* et *WTC*. Comme présenté dans le chapitre 3, *WTS* est la durée d'attente par l'agent DAN d'un message d'erreur après avoir lancé une demande de réservation de ressources pour un flux Premium. Une fois expiré, l'agent considère que sa demande a été admise et informe l'application pour commencer à émettre son trafic.

Le minuteur *WTC* est la durée maximale d'attente de rafraichissement de

réservation. Si la durée de *WTC* d'un flux Premium s'écoule sans réception d'un paquet data permettant de rafraichir sa réservation, le nœud considère que la réservation est annulée et libère les ressources.

Nombre de nœuds	10
Modèle de mobilité	PPRZM
Protocole de routage	AODV-UU
Protocole MAC	802.11
Capacité du canal	54 Mbits/s
Capacité maximale du trafic Premium	11 Mbits/s
Trafic Urgent/nœud	500 Octets : envoyé aléatoirement chaque [1s,3s]
Trafic Premium/nœud	760 Kbits/s : 950 Octets chaque 10 ms
Trafic Best-effort /nœud	4 Mbits/s 500 Octets chaque 1ms
Zone de simulation	600 m x 600 m
Portée radio	100m

TABLE 4.7 – Deuxième scénario

Pour ces deux études, nous avons utilisé le scénario présenté dans le tableau 4.7. Tous les nœuds émettent leurs trafics vers une seule destination qui est la station sol.

4.2.2.1 Le minuteur *WTC*

La valeur de *WTS* a été fixé à 1 sec durant cette simulation puisque cette valeur est suffisamment grande pour ne pas congestionner le réseau par des flux Premium non admis par le contrôleur d'admission ce qui est validé par l'application ciblée.

Le minuteur *WTC* est responsable de la mise à jour des données du système. Par conséquent, cette durée ne doit pas être grande puisque le système risque de rejeter des réservations de nouveau flux Premium en se basant sur des données qui ne sont pas mises à jour.

Nous avons fait varier la valeur *WTC* (1 sec, 1.5 sec, 2 sec, 3 sec, 5 sec, 10 sec et 20 sec) dans le but d'étudier l'influence de cette variation sur les performances du système. Ces valeurs sont choisies puisque dans notre scénario chaque drone échange 100 paquets Premium par seconde. La pertes de 100 paquets est suffisante pour fermer une réservation. En conséquence,

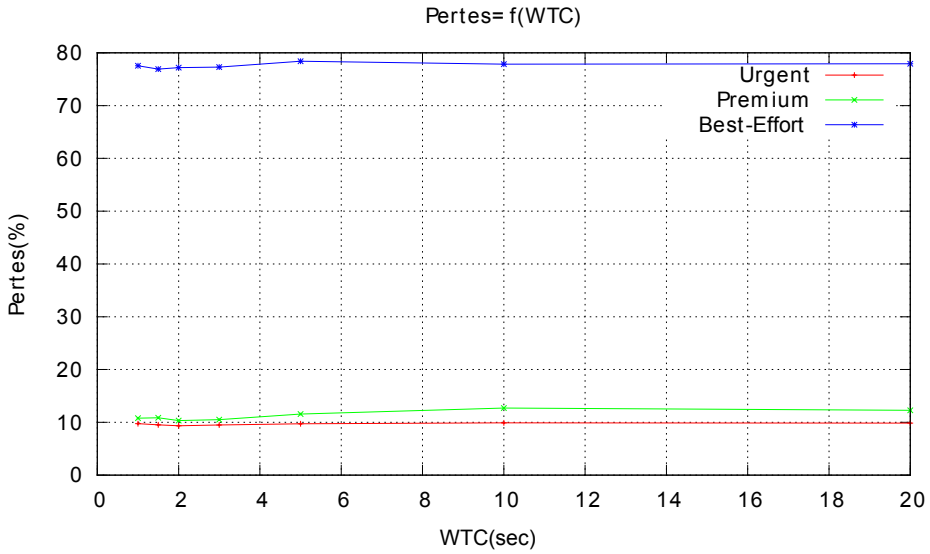


FIGURE 4.14 – Taux de perte mesurés avec chaque valeur de WTC pour les trois classes de trafic

Les durées fixées pour cette simulation sont suffisantes pour décider l'annulation des reservations.

Taux de perte

La figure 4.14 présente les taux de perte mesurés avec chaque valeur de WTC pour les trois classes de trafic.

Nous remarquons que la variation de la durée de WTC n'affecte pas les mesures de pertes des classe Urgent et Best-effort qui gardent pratiquement les mêmes valeurs avec les différents tests réalisés, contrairement aux pertes de la classe Premium qui augmente légèrement avec l'augmentation de la durée de WTC. En effet, durant la simulation les nœuds changent de positions, ainsi, leurs paquets suivent de nouvelles routes autres que celles déjà réservées. En gardant les données du réseau au niveau de chaque nœud durant une durée WTC importante, les nouveau voisins risquent de voir leurs demandes de réservation rejetées. Ce qui augmente le taux de perte de cette classe.

Délai de bout en bout

La figure 4.15 présente les délais de bout en bout mesurés avec les différentes valeurs de WTC pour les trois classes de trafic.

Les deux classes Urgent et Best-effort gardent les mêmes valeurs de délais

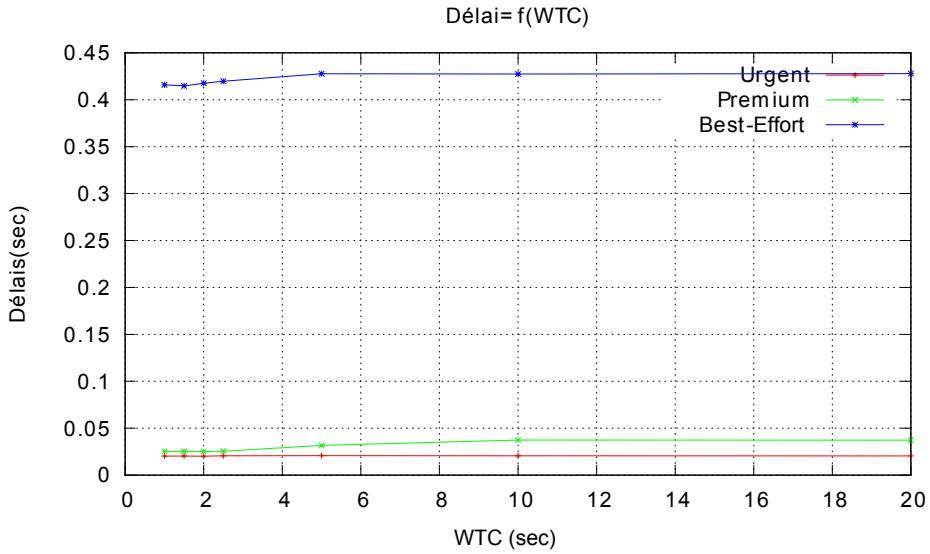


FIGURE 4.15 – Délais de bout en bout mesurés avec les différentes valeurs de *WTC*

avec les différentes simulations réalisées. Par contre, le délais de bout en bout de la classe Premium augmente légèrement avec l'augmentation de la durée de *WTC*.

Débit de transmission

D'après les courbes de la figure 4.16, qui présentent les moyennes de débits de transmission mesurées pour les classes Premium et Best-effort durant ces différentes simulations, la variation de la durée de *WTC*affecte le débit de la classe Premium qui diminue en augmentant cette durée. Cette diminution est causée par le rejet des réservations pour les flux Premium, ce qui diminue le débit du trafic Premium dans le réseau.

La signalisation échangée

Après l'expiration de *WTC*, le nœud annule la réservation et libère les ressources. Il existe deux causes pour que le minuteur *WTC* ne soit pas rafraichi :

- l'émetteur a terminé son émission mais le nœud n'a pas reçu le message `DAN_Close` ;
- à cause de la mobilité, l'émetteur s'éloigne du nœud et change de route vers la destination.

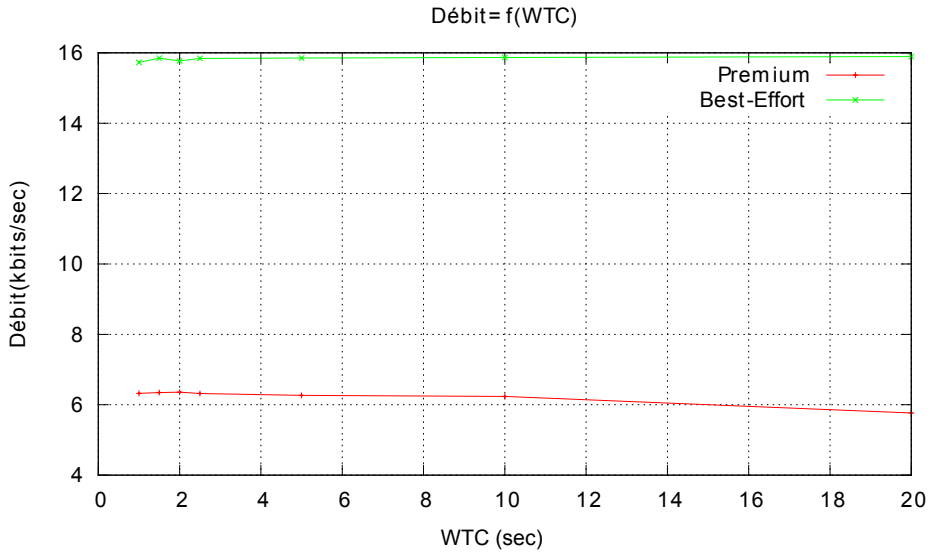


FIGURE 4.16 – Débits de transmission mesurés avec la variation de la durée WTC

Dans le deuxième cas, l'émetteur peut tenter d'utiliser l'ancienne route une autre fois. Dans ce cas, il y aura un échange de messages de signalisation entre les deux mobiles puisque à la réception des paquets de données d'un trafic Premium non admis par le contrôleur d'admission, le nœud doit informer la source par un message d'erreur pour qu'elle initialise la réservation.

Pour cette raison, nous avons étudié l'influence de la variation de la durée WTC sur le nombre de messages de signalisation échangés durant toute la simulation.

La figure 4.17 présente les rapports entre le nombre de paquets de signalisation envoyés durant toute une simulation et le nombre total de paquets Premium émis, mesurés avec chaque valeur de la durée de WTC. Autrement dit, cette figure présente le nombre de paquets de signalisation par un paquet de données Premium envoyé.

La courbe montre que, lorsque la durée de WTC augmente, le nombre de paquets de signalisation diminue légèrement, ce se qui s'explique par le fait que, lorsque on augmente la durée de WTC, les réservations sont maintenues pour des durées plus longues au cours desquelles des mobiles peuvent changer rapidement de position. Par conséquent, il y a moins d'échange de message de signalisation dans le réseau.

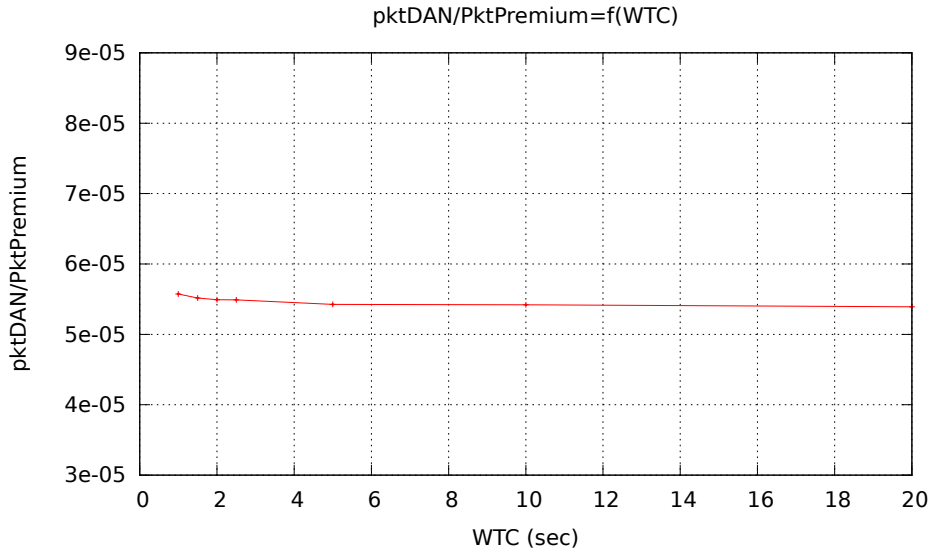


FIGURE 4.17 – Rapports entre le nombre de paquets de signalisation envoyés et le nombre total de paquets Premium émis mesurés avec les différentes durées de *WTC*

Cette étude montre que la durée du minuteur *WTC* affecte les performance du trafic Premium. Cette valeur ne doit pas être importante afin de garantir une mise à jour suffisamment fréquente du système. De plus, elle ne doit pas être très faible pour diminuer les échanges de signalisation dans le réseau. Pour la suite de notre étude, nous fixons la valeur de *WTC* à 2 sec puisque les résultats montrent que cette valeur a donné de meilleurs résultats par rapport aux autres durées étudiées.

4.2.2.2 Le minuteur *WTS*

Cette étude a pour but de borner la durée du minuteur *WTS*. Cette valeur doit être inférieure à celle du *WTC*, sinon les nœuds sources risquent de commencer à émettre leurs premiers paquets Premium après l'expiration des réservations. La valeur de *WTC* est fixée à 2 sec pour varier la valeur de *WTS* (0.01 sec, 0.1 sec, 0.5 sec, 1 sec, 1.5 sec et 2 sec) et étudier l'impact de cette variation sur les performances du système et pouvoir choisir la durée optimale.

Taux de perte

La figure 4.18 présente les taux de perte mesurés pour chaque classe de trafic avec les différentes durées de *WTS*.

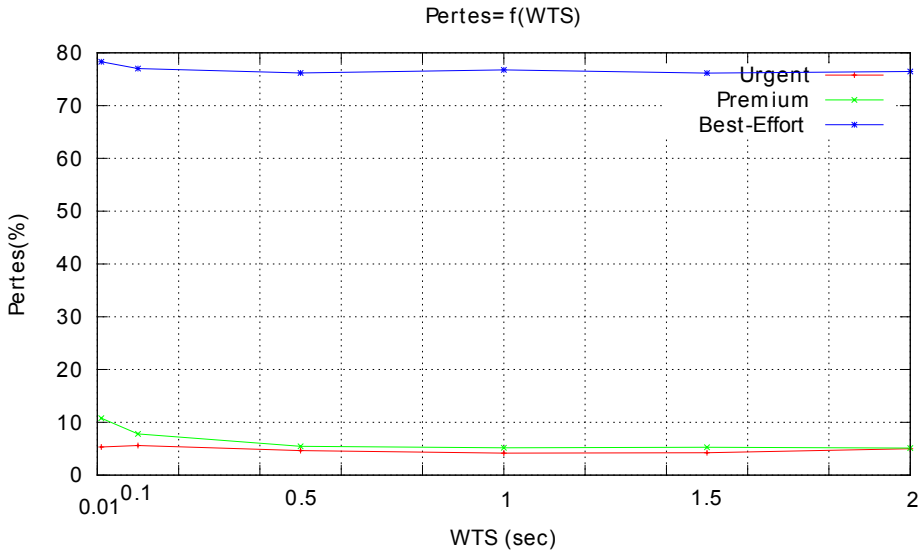


FIGURE 4.18 – Taux de perte mesurés pour chaque classe de trafic durant les différents tests

Nous remarquons une légère amélioration au niveau du taux de perte de la classe Premium dé 0.1 sec. Cela est dû au fait qu'après l'expiration de *WTS*, le nœud commence directement à émettre le trafic. Avec une très faible durée de *WTS*, le nœud peut commencer à émettre son trafic même avant l'établissement de la réservation ce qui augmente les risques de traiter les paquets déjà envoyés comme un trafic Best-effort.

Délai de bout en bout

La figure 4.19 présente les délais de bout en bout mesurés pour chaque classe de trafic avec les différentes durées de l'intervalle *WTS*.

La figure montre une légère amélioration au niveau du délai mesuré pour la classe Best-effort puisque, durant l'intervalle *WTS*, le système a moins de paquets Premium à émettre, ce qui diminue la charge des files d'attente et par conséquent, améliore le temps d'attente des paquets Best-effort dans leur file.

Débit de transmission

La figure 4.20 présente les moyennes des débits mesurés pour les deux classes de trafic Premium et Best-effort avec les différentes durées de l'intervalle *WTS*.

Nous remarquons une légère amélioration pour le débit de la classe Best-

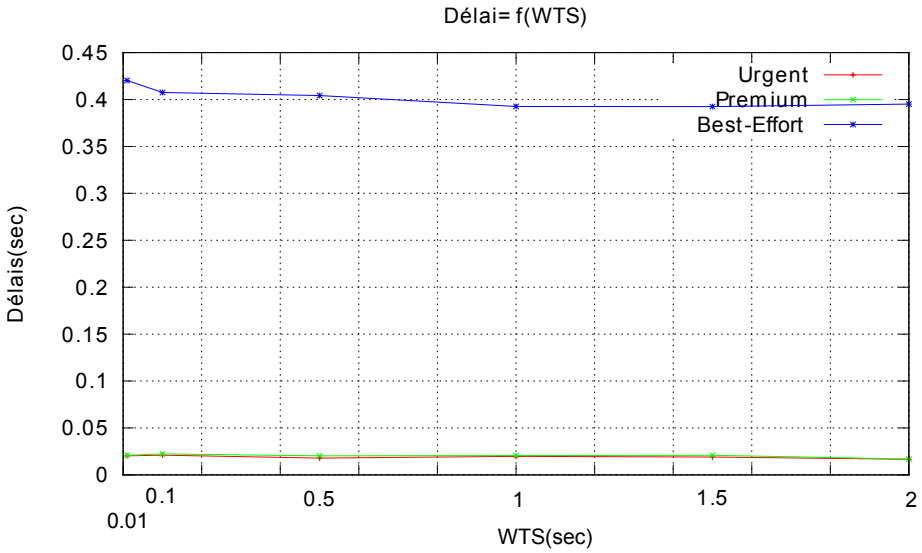


FIGURE 4.19 – Délais mesurés pour chaque classe de trafic durant les différentes simulations

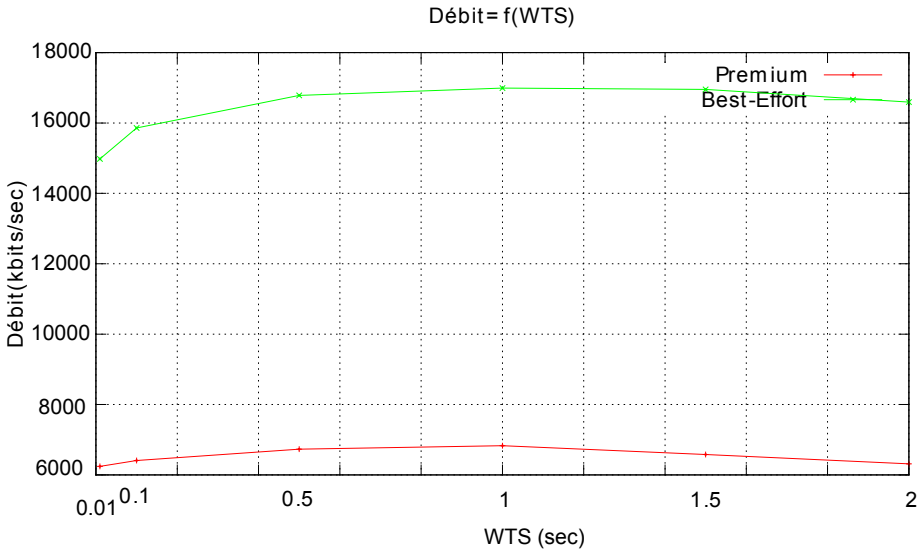


FIGURE 4.20 – Débit de transmission mesurés pour les différentes classes de trafic

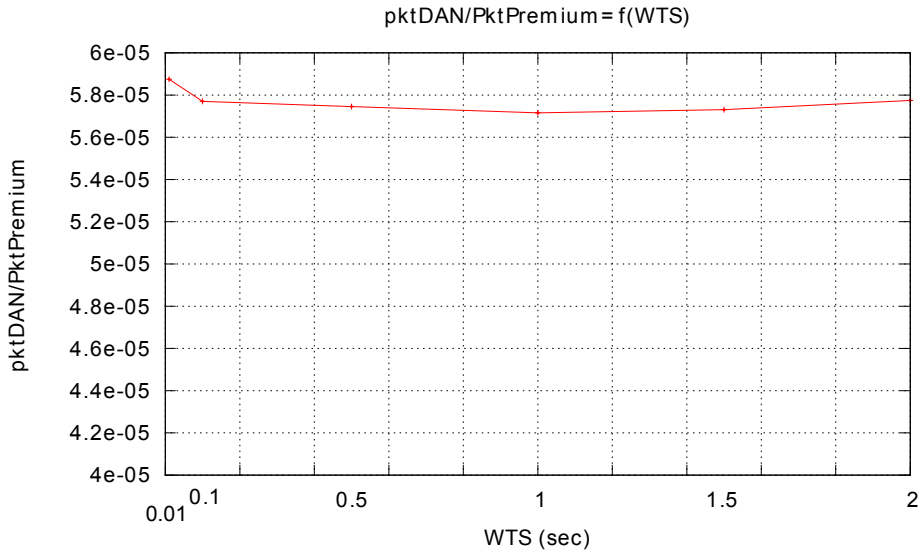


FIGURE 4.21 – Nombre de messages de signalisation échangés au cours de chaque simulation

effort due aux mêmes causes expliquées précédemment.

Trafic de signalisation

La figure 4.21 présente la variation du rapport entre le nombre de paquets de signalisation émis et le nombre de paquets de données Premium envoyés en fonction de la variation de la durée *WTS*.

La courbe montre une diminution du nombre de messages de signalisation envoyés durant la simulation (pour les valeurs de 0,1 sec $\leq$ *WTS* $\leq$ 2 sec), ce qui s'explique par la fait qu'avec une courte durée de *WTS*, la source peut commencer à émettre ses paquets Premium avant la réception du message d'erreur `DAN_Error`. En conséquence, le nombre de messages de signalisation augmente pour la ré-émission du message d'erreur. Avec les durées plus longues de *WTS*, les paquets risquent de traverser de nouvelles routes autres que celles déjà réservées, ce qui oblige les nœuds à échanger davantage de messages de signalisation.

Cette étude permet de borner la durée de l'intervalle *WTS*. Cette valeur doit être entre 0,1 sec et la durée de *WTC*. Pour le reste de notre étude, nous fixons la durée de *WTS* à une seconde, puisque cette valeur est supérieure à la durée du délai d'aller-retour dans le réseau (la valeur moyenne calculée est égal à 275ms) et inférieure à la valeur *WTC* déjà choisie. De plus, elle

est validée par l'application ciblée puisque nous remarquons sur les courbes présentées précédemment (figures 4.20 et 4.21) que cette valeur réalise les meilleurs résultats en termes de débit et de la signalisation échangée par rapport aux autres durées.

4.2.3 Performances générales de DAN

Cette étude a pour but d'évaluer les performances de DAN dans un réseau congestionné. Le tableau 4.8 présente le scénario de cette simulation.

Nombre de nœuds	11
Modèle de mobilité	PPRZM
Protocole de routage	AODV-UU
Protocole MAC	802.11
Capacité du canal	54 Mbits/s
Capacité maximale du Premium	11 Mbits/s
Trafic Urgent/nœud	500 Octets : envoyés aléatoirement chaque [1s,3s]
Trafic Premium/nœud	760 Kbits/s : 950 Octets chaque 10 ms
Trafic Best-effort /nœud	4 Mbits/s : 500 Octets chaque 1ms
Zone de simulation	600 m x 600 m
Portée radio	100m

TABLE 4.8 – Troisième scénario

Tous les nœuds émettent le même trafic vers une seule station sol.

Taux de perte

Le tableau 4.9 présente les taux de perte calculés pour chaque classe de trafic.

	Urgent	Premium	Best-effort
Pertes dues à la congestion	0	0	74.678
Pertes totales	7.79	8.04	76.614

TABLE 4.9 – Taux de perte mesuré pour chaque classe de trafic (%)

Les résultats montrent une grande différence entre les pertes de la classe Best-effort et celles des classes Urgent et Premium. En effet, les mécanismes

de gestion de trafic retardent l'émission des messages Best-effort afin de garder la priorité des autres classes. Ainsi, ces paquets saturent la file d'attente qui commence à les rejeter.

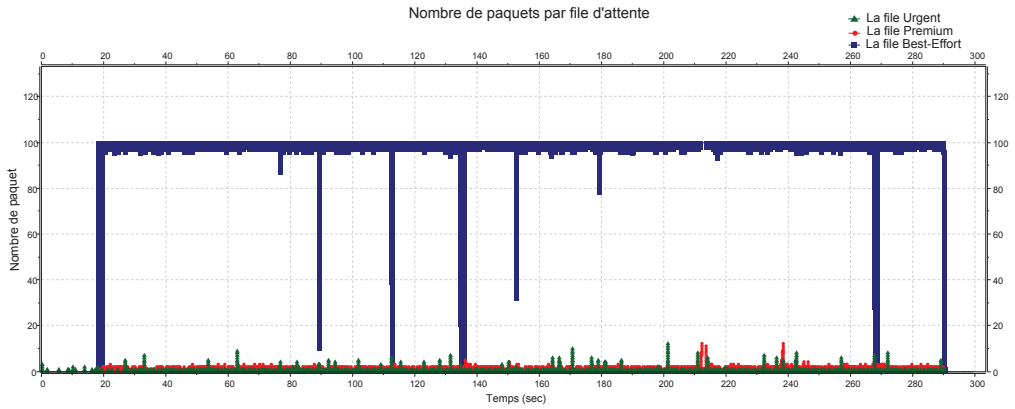


FIGURE 4.22 – Nombre de paquets par chaque file d'attente

La figure 4.22 présente le nombre de paquets par file d'attente obtenue au cours de la simulation. Elle confirme que la file Best-effort était saturée durant toute la durée de la simulation, ce qui explique que le taux de perte dû à la congestion de la classe Best-effort est égal à 74,67% alors qu'il est nul pour les trafics Urgent et Premium.

Malgré la gestion de trafic de DAN, les résultats montrent des pertes au niveau des classes Urgent et Premium (7,69% et 8,14% ainsi que 76,61% pour la classe Best-effort). Ces pertes sont causées par des problèmes liés au réseau comme des interférence et des collisions.

Délai de bout en bout

Le tableau 4.10 présente les délais de bout en bout mesurés pour chaque classe de trafic. Il montre l'impact de la gestion de la QdS sur le trafic Best-effort qui réalise un délais de 397 ms contre 20 ms et 28 ms pour les deux classe Premium et Urgent.

	Urgent	Premium	Best-effort
Délai de bout en bout	0.016	0.02	0.397

TABLE 4.10 – Délai de bout en bout mesuré pour chaque classe de trafic

La saturation du canal de transmission a un effet aussi sur l'augmentation des délais. En effet, chaque nœud doit attendre son rôle pour pouvoir émettre ses paquets ce qui peut tarder lorsque le canal partagé est saturé.

Débit de transmission

La figure 4.23 et le tableau 4.11 présentent les débits mesurés pour les deux classes de trafic Premium et Best-effort. Pour cette étude, le débit est mesuré chaque 0,1 seconde. Les valeurs présentées sont les moyennes sur des intervalles de 1 seconde.

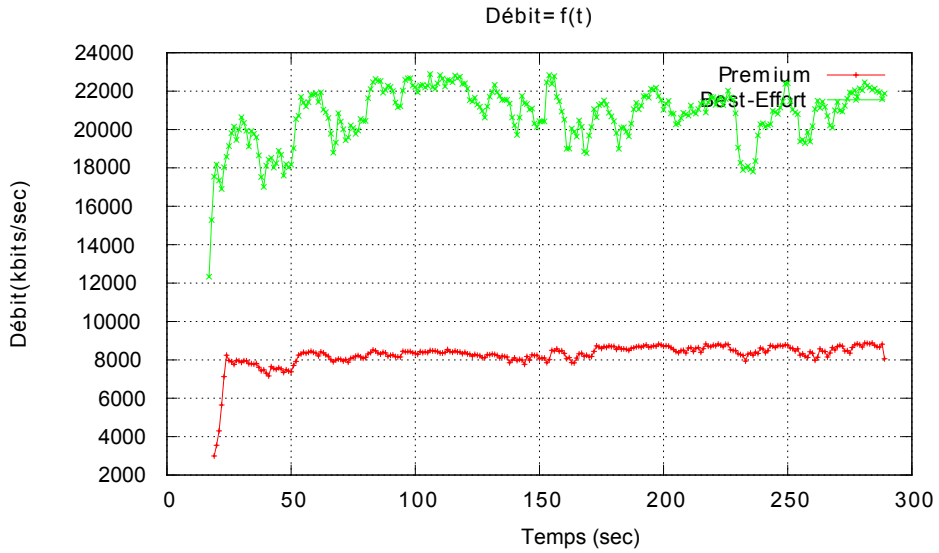


FIGURE 4.23 – Débit mesuré durant la simulation pour les deux classes Premium et Best-effort

	Min	Max	Moyenne
Premium	2993	8876	8268,449
Best-effort	12332,31	22888	20762,538

TABLE 4.11 – Débit mesuré pour les deux classes de trafic Premium et Best-effort (Kbits/s)

La courbe montre que le système a réussi à offrir le débit demandé par le trafic Premium (670 Kbits/s par nœud). Nous remarquons aussi que le débit du trafic Best-effort a subi plus de fluctuations que le débit du trafic Premium qui a gardé les mêmes valeurs durant toute la simulation. Cela montre que DAN garantit le débit demandé par les flux Premium.

Cette étude montre l'efficacité de DAN dans le cas des réseaux congestionnés puisqu'il respecte les QdS demandées par chaque classe de trafic. Il offre des taux de perte et des délais plus faibles pour le trafic Urgent et garantit le débit optimal demandé par les flux Premium.

Malgré l'efficacité de DAN à différentier le trafic et à gérer la QdS dans les situations de congestion, il ne peut rien faire face aux collisions et aux interférences qui ont causé quelques pertes au niveau des deux classes Premium et Urgent.

4.2.4 Évaluation de l'impact de variation de la charge du réseau

Cette étude a pour but d'évaluer les performances de DAN en augmentant à chaque fois le nombre d'émetteurs dans un réseau à multi-saut.

Nombre de nœuds	10
Protocole de routage	AODV-UU
Protocole MAC	802.11
Capacité du canal	54 Mbits/s
Capacité maximale du Premium	11 Mbits/s
Trafic Urgent/nœud	500 Octets : envoyés aléatoirement chaque [1s,3s]
Trafic Premium/nœud	760 Kbits/s : 950 Octets chaque 10 ms
Trafic Best-effort /nœud	4 Mbits/s : 500 Octets chaque 1ms
Zone de simulation	600 m x 600 m
Portée radio	100m

TABLE 4.12 – Quatrième scénario

Le tableau 4.12 présente le scénario de cette simulation. Le nombre d'émetteurs varie avec chaque test dans le but de modifier la charge du réseau et ils émettent tous leurs trafics vers une seule station de contrôle.

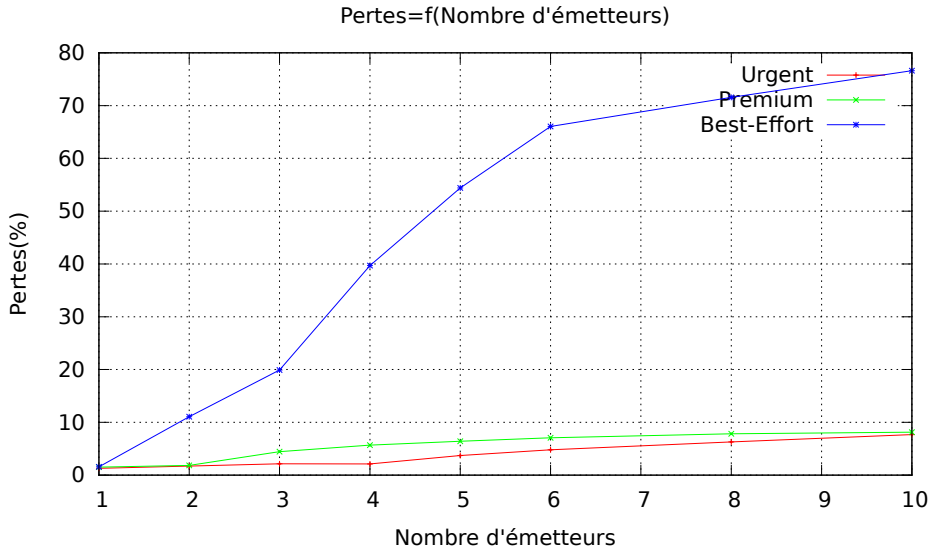


FIGURE 4.24 – Taux de perte mesuré pour chaque classe de trafic

Taux de perte

La figure 4.24 présente les taux de perte de chaque classe de trafic mesurés avec les différents nombres d'émetteurs. La courbe montre une différence remarquable entre le taux de perte de la classe Best-effort et ceux des autres classes. De plus, les pentes des courbes sont totalement différentes. En effet, le taux de perte des deux classes de trafic Premium et Urgent augmente très lentement contrairement à celui du trafic Best-effort qui augmente rapidement.

Ces résultats montrent que, même quand la charge dans le réseau augmente, DAN garantit un taux de perte très faible pour les deux classes prioritaires Urgent et Premium.

Délai de bout en bout

La figure 4.25 présente les délais de bout en bout mesurés pour chaque classe de trafic avec les différents nombres d'émetteurs.

Pour des réseaux peu chargés, (1 ou 2 émetteurs), les résultats ne montrent pas une différence de délais entre les trois classes. Pourtant, à chaque fois que la charge augmente dans le réseau, le délai de bout en bout de la classe Best-effort augmente rapidement alors que les deux autres classes évoluent lentement en gardant les valeurs du délai les plus faibles. En effet, plus il y a des paquets Premium et Urgent qui circulent dans le réseau plus la durée d'attentes des paquets Best-effort dans leur file d'attente augmente.

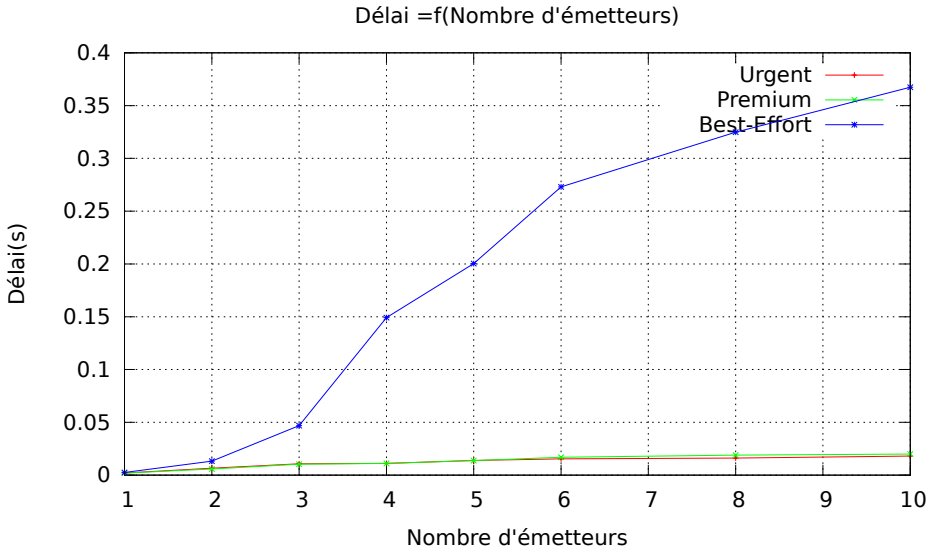


FIGURE 4.25 – Délai mesuré pour chaque classe de trafic

Ces résultats montrent que DAN garantit un délai de bout en bout très faible pour les deux classes de trafic Premium et Urgent comme requis.

Débit de transmission

La figure 4.26 présente les valeurs moyennes de débit de transmission mesurées pour les deux classes Premium et Best-effort en faisant varier le nombre d'émetteurs dans le réseau.

Elle montre que le débit de la classe Premium augmente d'une manière linéaire avec le nombre d'émetteur, ce qui montre que le système est capable de fournir le débit nécessaire pour ces flux. Cette augmentation ne menace pas le débit du trafic Best-effort grâce au seuil fixé par le contrôleur d'admission. En effet, comme dans l'étude 4.2.1 où les flux Premium dépassent la capacité totale allouée à cette classe, le système n'admet les demandes de réservation que dans les limites de la capacité totale allouée au trafic Premium. Le reste du trafic est traité comme Best-effort.

Cette étude montre que DAN est capable de s'adapter à la charge du réseau et de garantir une différentiation de services. Il fournit les QdS demandées en termes de délais de bout en bout et de taux de perte pour les messages Urgent ainsi que de garantir le débit de transmission requis pour chaque flux Premium. Ces QdS sont respectées même en variant la charge dans un réseau à multi-saut des nœuds mobiles.

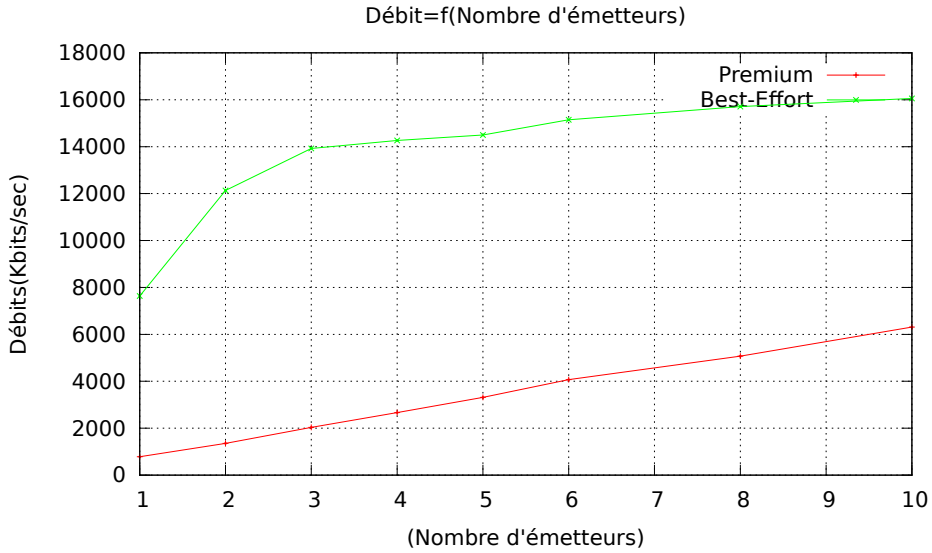


FIGURE 4.26 – Moyennes de débit de transmission mesurées pour les deux classes de trafic Premium et Best-effort

4.2.5 Évaluation avec d' autres protocoles de routage

L'une des exigences de conception de DAN est de fonctionner indépendamment du protocole MAC et du protocole de routage ce qui permet de l'adapter à n'importe quel système d'agents coopératifs.

Cette étude a pour but d'évaluer le fonctionnement de DAN avec des protocoles de routage autres que AODV pour valider cette exigence.

Pour cette raison, nous avons lancé le même scénario que l'étude 4.2.3 mais en utilisant deux autres protocoles de routage : OLSR et DSDV qui sont des protocoles de routage pro-actifs.

Taux de perte

La figure 4.27 présente les taux de perte mesurés pour chaque classe de trafic avec les deux protocoles de routage.

Le graphe montre une différence remarquable entre les pertes de la classe Best-effort (78,88% et 77,51%) et celles des classes Premium et Urgent (entre 10,21% et 4,28% pour la classe Urgent et 10,62% et 6,9% pour les flux Premium).

Ces résultats montrent que DAN permet d'obtenir les taux de perte les plus faibles pour les classes Urgent et Premium indépendamment du protocole de routage.

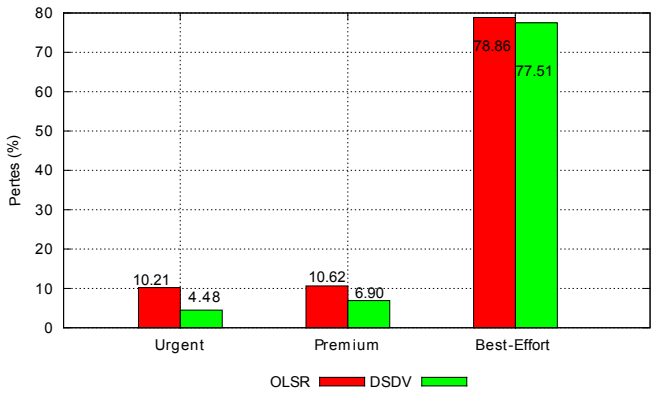


FIGURE 4.27 – Taux de perte mesurés pour les différentes classes de trafic

Délai de bout en bout

La figure 4.28 présente les délais de bout en bout mesurés pour les différentes classes de trafic avec les deux protocoles de routage OLSR et DSDV.

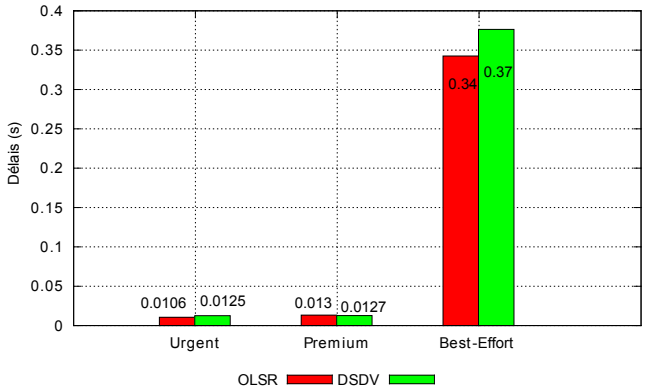


FIGURE 4.28 – Délai mesuré pour chaque classe de trafic

Elle montre la grande différence entre les délais mesurés pour le trafic Best-effort (0,34 sec et 0,37 sec) et ceux mesurés pour les deux autres classes de trafic (environ 0,012 sec).

Ces résultats confirment que DAN offre le délai le plus faible pour les trafics Urgent et Premium indépendamment du protocole de routage.

Débit de transmission

La figure 4.29 présente les moyennes des valeurs de débits mesurées pour les classes de trafic Premium et Best-effort avec les deux protocoles de routage OLSR et DSDV.

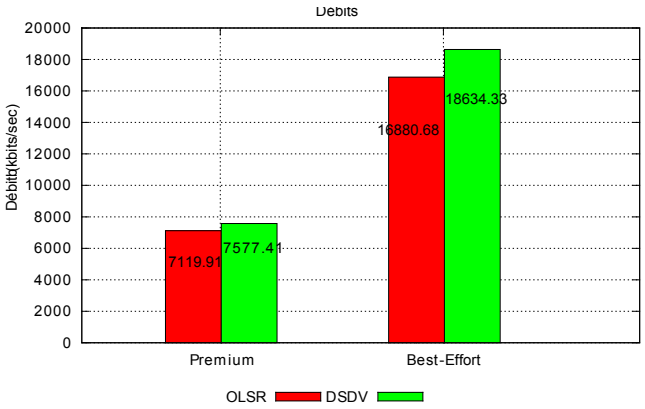


FIGURE 4.29 – Débits mesurés pour les deux classes Best-effort et Premium avec les protocoles de routage OLSR et DSDV

Les résultats montrent que DAN respecte le débit demandé par les flux Premium avec les différents protocoles de routages.

Cette étude montre que DAN peut s'adapter à n'importe quel protocole de routage pour différencier le trafic et garantir la QdS demandée pour chaque classe.

Conclusion

Ce chapitre a présenté l'étude de performance de DAN par simulation. Cette étude a été réalisée dans un environnement proche de la réalité en utilisant le modèle de mobilité PPRZM.

PPRZM reproduit les mouvements réels des drones Paparazzi d'une manière aléatoire. Il met en œuvre certains types de mouvements qui ne sont pas pris en compte par les autres modèles de mobilité usuels comme le mouvement de balayage d'une zone géographique ou le fait de refaire le même mouvement dans la même zone durant un intervalle de temps.
Nous avons choisi d'utiliser ce modèle de mobilité puisque dans le cas d'une flotte de drones, l'obtention des traces de plusieurs aéronefs est difficile à cause du nombre limité de drones et la complexité de lancement des vols. PPRZM offre la possibilité de faire varier les paramètres de simulation comme la durée du scénario ou le nombre de drones utilisés, dans le but d'étudier une diversité de cas possibles. Par conséquent, nous avons pu créer des scénarios de simulation difficiles à réaliser avec des drones réels (comme le nombre élevé de drones par test).

Les résultats de l'étude de DAN montrent qu'il différentie le service en trois classes : Urgent, Premium et Best-effort et qu'il garantit différent niveau de QdS. En effet, Il offre le taux de perte et le délai les plus faibles pour les messages Urgent et garantit le débit de transmission demandé pour chaque flux Premium même dans les réseaux congestionnés.

Le chapitre suivant présente l'implémentation de DAN sur des drones réels ainsi que les expérimentations réalisées.

Chapitre 5

Implémentation et plate-formes de tests

Contents

5.1 Outils utilisés 118

5.2 Implémentation de DAN 119

5.3 Résultats et discussion 121

5.3.1 Validation du fonctionnement du nouveau système de communication Paparazzi 123

5.3.2 Évaluation de DAN 127

Introduction

Comme présenté dans le Chapitre 1, les communications entre les drones Paparazzi et la station sol étaient établies à travers un module XBee 802.15.4 supportant les communications point à point et point à multi-points. Sur la liaison montante (de la station sol (GCS pour *Ground Control Station*) vers les drones), les messages sont émis en diffusion. Par la suite, selon le contenu, c'est à chaque drone de décider s'il le traite ou non. Sur la liaison descendante (du drone vers l'unique station de contrôle), les messages étaient envoyés en unicast.

Ce chapitre présente l'opération d'intégration de DAN au système Paparazzi dans les deux premières sections. Par la suite, les différentes expérimentations réalisées ainsi que leurs résultats sont détaillés dans la troisième section.

5.1 Outils utilisés

L'intégration est réalisée sur des quadricoptères Paparazzi (figure 5.1). Ces drones ont été modifiés en ajoutant une carte Raspberry Pi [1] qui utilise la distribution Linux Raspbian [2], ce qui nous a permis d'utiliser une implémentation existante du protocole de routage AODV-UU [GSB09] pour Linux.

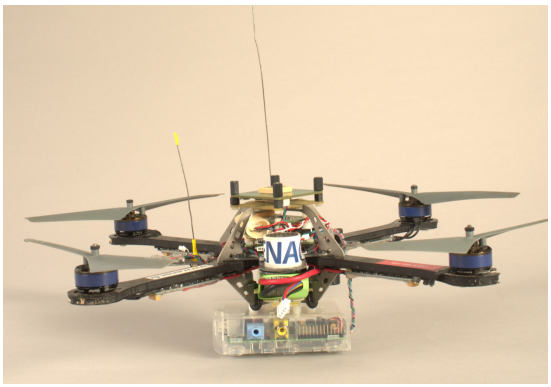


FIGURE 5.1 – Quadricoptère Paparazzi

Afin de réaliser ce système, deux modifications majeures ont été nécessaires, une au niveau logiciel et l'autre au niveau matériel.

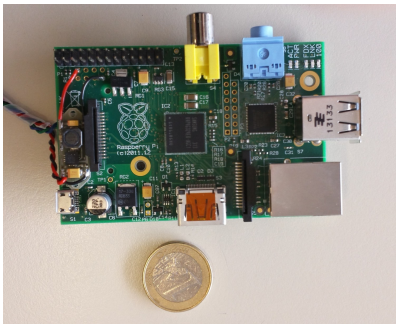


FIGURE 5.2 – Carte Raspberry Pi

Du point de vue matériel, plusieurs modifications ont été réalisées. En effet, les cartes Raspberry, avec leurs boitiers, ont été attachées à chaque quadricoptère (comme sur la figure 5.1 où on voit le quadricoptère Paparazzi qui se pose sur le boitier de la carte Raspberry Pi). Pour cela, il était

1. http ://www.raspberrypi.org/
2. http ://www.raspbian.org/

nécessaire de modifier la carte mémoire SD en micro-SD afin de garantir un meilleur mécanisme de verrouillage. Le but de cette modification est d'éviter la perte de la liaison avec la carte causées par des vibrations durant le vol. La carte Raspberry est alimentée directement à partir de la batterie du drone.

Au niveau MAC et physique nous utilisons des clés *Wi-Pi* (figure 5.3) qui offrent une connexion avec la norme IEEE 802.11g. Son canal de transmission a une capacité théorique de 54 Mbits/s et permet de connecter des mobiles à une portée de 75 m qui peut s'élever à environ 200 m sur une vision directe.

FIGURE 5.3 – Une clé Wi-Pi

D'autres logiciels ont été développés permettant d'agir comme une interface entre le système de communication traditionnel de Paparazzi et les paradigmes du réseau ad hoc (adressage IP) et de passer du principe de diffusion sur la liaison montante et unicast sur la liaison descendante en unicast sur les deux liaisons.

La figure 5.4 présente le nouveau système Paparazzi. Dans ce système, DAN est utilisé au niveau de chaque drone pour assurer les communications entre eux et avec la station sol à travers un réseau ad hoc.
Au niveau de la station de contrôle, DAN est implémenté sur une carte *Raspberry* connectée au module *Link* de la station à travers une liaison série. Ce module, qui assure les communications avec le réseau, est lié avec les autres fonctionnalités de la station de contrôle, comme *SRV* (le serveur de données), à travers un bus *IVY*[3]. IVY est un bus logicielle permettant de communiquer des processus en différents langage (C, C++, Java, Python, Perl, Unix et Mac OS).

5.2 Implémentation de DAN

DAN est implémenté en C++, dans l'espace utilisateur du système linux en utilisant la librairie *libnl* (Netlink Protocol Library) (figure 5.5) ainsi que

3. http ://www.eei.cena.fr/products/ivy/

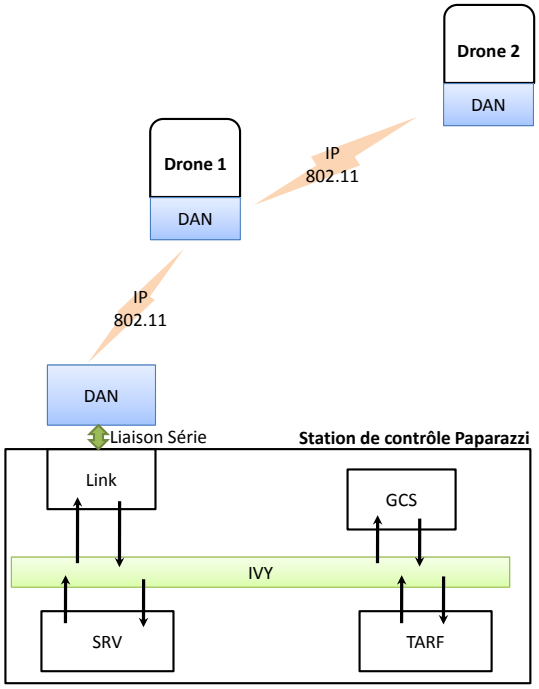


FIGURE 5.4 – l'intégration de DAN au système Paparazzi

les mécanismes du système Linux permettant de gérer la QdS [lin].

Cette librairie permet de gérer les fonctionnalités du noyau à partir de l'espace utilisateur. *Libnl* met en œuvre des principes fondamentaux nécessaires pour utiliser le protocole *Netlink* [SKKK03] qui permet de communiquer le noyau et l'espace utilisateur, essentiellement pour échanger des données sur le réseau et permet de réaliser des fonctionnalités comme la construction des messages, l'envoi et la réception de données. La commande *Iproute* est un exemple des utilités qui utilisent le protocole *Netlink*.

De plus *Libnl* offre des moyens permettant de gérer la plateforme *Netfilter*. *Netfilter* est une plateforme du noyau Linux qui offre une variété d'options pour manipuler la réception et l'émission des paquets du réseau. Autrement dit, *Netfilter* permet de filtrer les paquets reçus ou à émettre.

Afin de simplifier l'utilisation du standard C++ et les communications entre les différents processus du système nous avons utiliser la librairie *boost*[4].

La figure 5.5 schématise l'architecture du système Linux contenant DAN. Elle montre que dans l'espace utilisateur, les applications communiquent avec

4. http ://www.boost.org/

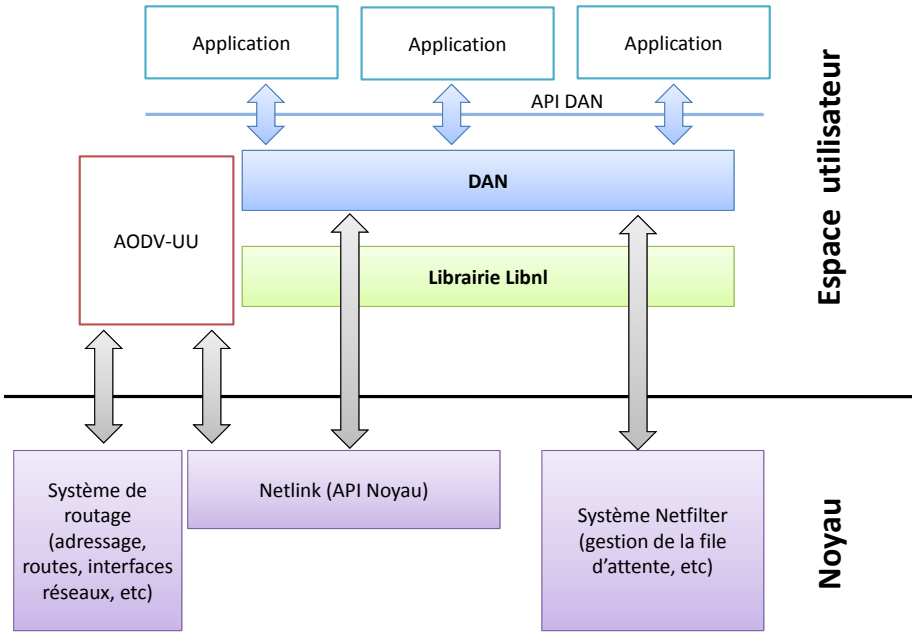


FIGURE 5.5 – Architecture du système Linux avec DAN

DAN à travers des *sockets* de l'API DAN. DAN, implémenté dans l'espace utilisateur, communique avec le noyau en utilisant la librairie *Libnl* dans le but de gérer les plate-formes *Netfilter* et *Netlink*. Le protocole de routage AODV-UU, implémenté aussi dans l'espace utilisateur, gère les fonctionnalités de routage du noyaux Linux.

La figure 5.6 présente un extrait du diagramme de classe du l'agent DAN. Elle schématise les relations entre la classe Agent DAN et les autres classes comme les classes `DAN_Msg` ou `DAN_Flow`.

5.3 Résultats et discussion

Cette section présente les résultats des expérimentations réalisées pour valider et évaluer les performances du nouveau système de communication de Paparazzi. Elle est divisée en deux parties : la première a pour but la validation du système de communication ad hoc (les modifications réalisées, le routage implémenté, etc) et la deuxième sert à évaluer les performances de DAN dans un environnement réel.

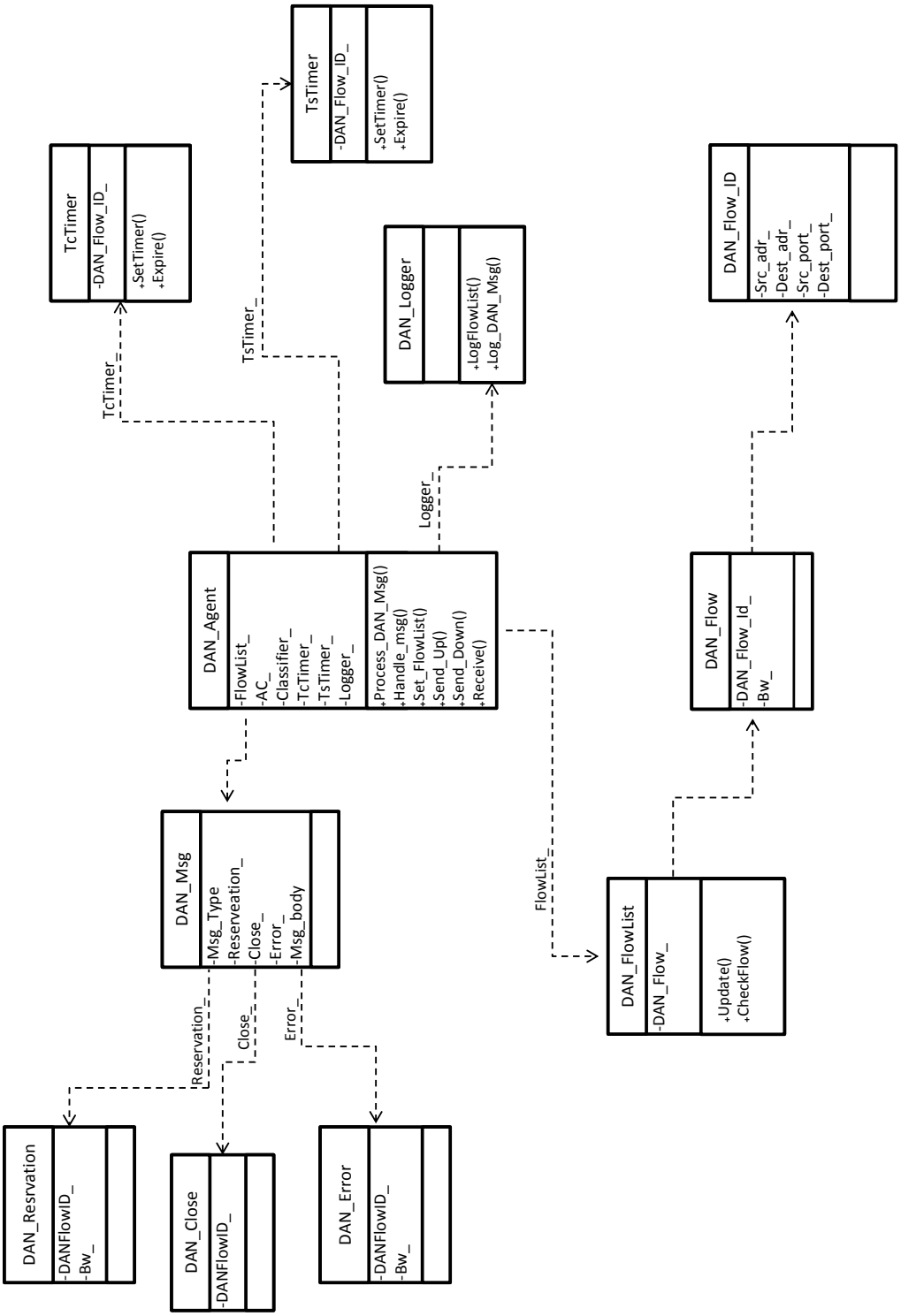


FIGURE 5.6 – Extrait du diagramme de classe de l'agent DAN

5.3.1 Validation du fonctionnement du nouveau système de communication Paparazzi

Les Premiers tests réalisés avaient pour but de montrer la possibilité de contrôler le système Paparazzi à travers un réseau ad hoc. Pour cela, nous avons utilisé deux drones. Ces drones se déplacent selon la trajectoire présentée dans la figure 5.7.

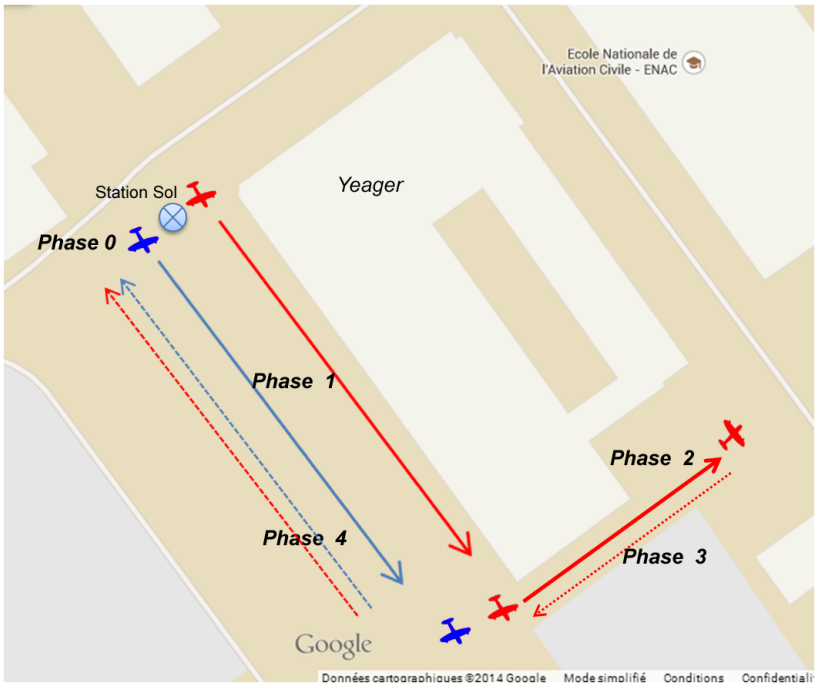


FIGURE 5.7 – Trajectoire des drones au cours de l'expérimentation

L'expérimentation est réalisée sur cinq phases autour du bâtiment *Yeager* de l'ENAC (de longueur 85 m). A l'instant t_0, les deux drones restent stationnaires à coté de la station sol durant environ 30 secondes (Phase 0). Par la suite, ils avancent ensemble dans la même direction (Phase 1). Une fois arrivé à l'extrémité, l'un des deux drones reste stationnaire dans sa position alors que l'autre continu son chemin derrière le bâtiment afin de perdre le lien direct entre lui et la station sol (Phase 2). Cette coupure de liaison force le réseau à router les paquets à travers le deuxième aéronef.
Ce scénario est choisi pour pouvoir perdre la connexion entre la station sol et l'un des drones. En effet, la portée des cartes MAC utilisée est de l'ordre de 200 m en vision directe. Ainsi, nous avons choisi d'utiliser le bâtiment comme obstacle.

En arrivant à l'autre extrémité du bâtiment, le drone reste stationnaire dans sa position durant 30 secondes avant de faire un demi-tour et revenir à coté de l'autre aéronef (Phase3). Par la suite, les deux drones reviennent à leurs positions initiales à coté de la station sol.

En plus de trafic Paparazzi échangé entre les drones et la station sol nous avons utilisé au niveau de chaque machine : `ping` pour mesurer le délai du réseau et `tcpdump` pour enregistrer le trafic. Ces captures permettent de calculer le délai, le taux de perte ainsi que le débit de transmission dans le réseau.

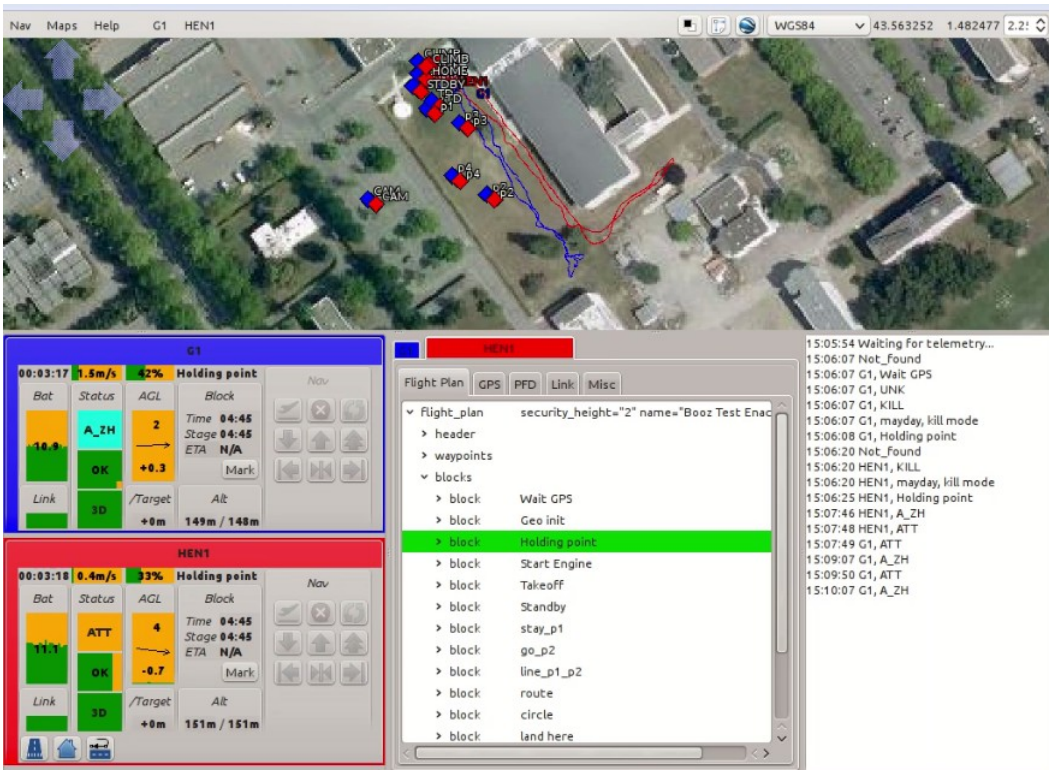


FIGURE 5.8 – Capture d'écran de la station de contrôle

La figure 5.8 est une capture d'écran de la station de contrôle réalisée à la fin de l'expérimentation. Elle permet de distinguer les deux trajectoires suivies des deux drones nommés *G1* (en bleu) et *HEN1* (en rouge).

Tous les résultats présentés dans cette section partagent le même axe temporaire qui commence avec le lancement de `tcpdump` au niveau de la station sol.

Après 40 secondes, les drones commencent à bouger. Ils atteignent l'extrémité

du bâtiment à t= 100 secondes et autour de 140 secondes, HEN1 arrive à la position la plus éloignée de sa trajectoire. IL revient à coté du drone G1 à l'instant 160 secondes. Ces informations, obtenues de la station sol qui enregistre toutes les données de vol, sont cohérentes avec les mesures réalisées pour le réseau comme présenté dans les courbes de la figure 5.9.

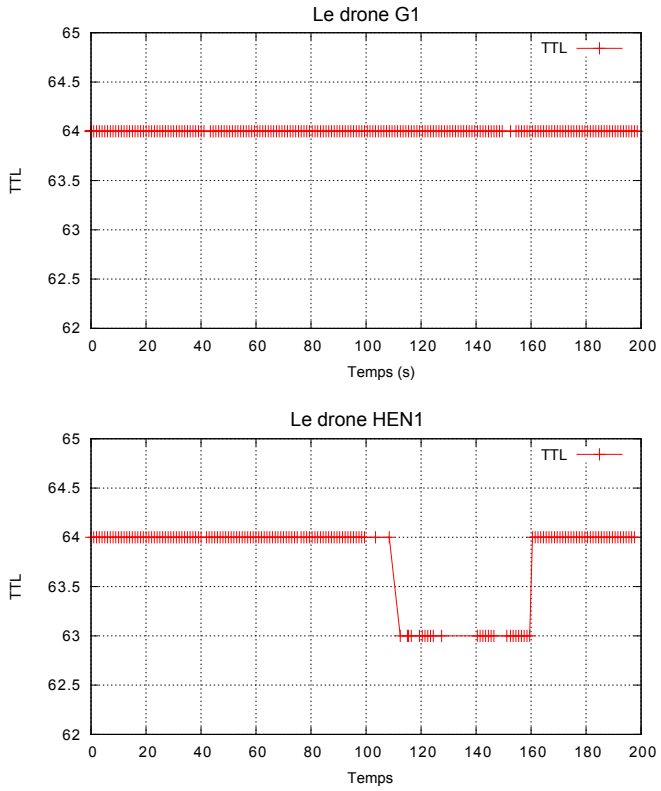


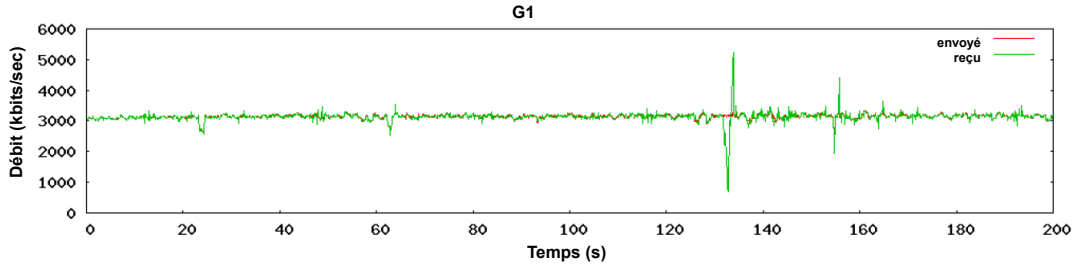
FIGURE 5.9 – Les valeurs TTL des paquets des deux drones G1 et HEN1

Les courbes présentent le contenu du champ TTL (*Time To Live*). Les paquets transmis sans routage ont une valeur TTL de 64 alors que les autres voient diminuer cette valeur avec chaque passage à travers un nœud intermédiaire. Le routage au niveau de G1 est réalisé entre 111 secondes et 159 secondes puisque durant cet intervalle de temps la valeur de TTL passe à 63.

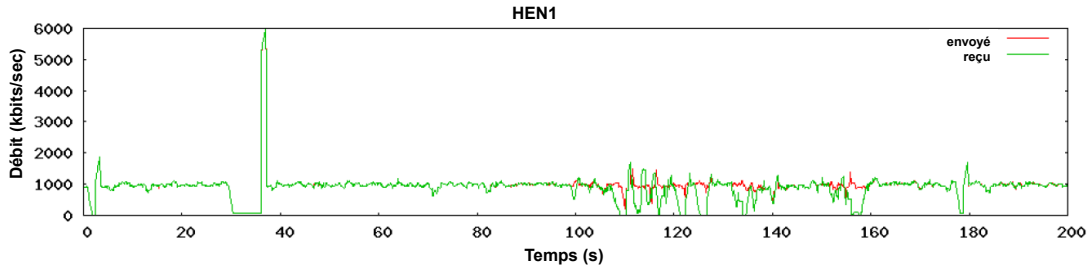
Débit de transmission

Le débit est mesuré pour le trafic descendant envoyé de chaque drone vers la station sol.

Les courbes de la figure 5.10 présentent le débit de trafic envoyé par



(a)



(b)

FIGURE 5.10 – Débit du trafic descendant

chaque drone, mesuré à l'émission, au niveau de chaque drone, et à la réception, au niveau de la station sol. Le débit est mesuré chaque 0,1 seconde, par la suite leur moyennes sur des intervalles de 1 secondes sont calculées.

Le tableau 5.1 présente les valeurs minimales, maximales et moyennes des débits mesurés au niveau de chaque drone et de la station sol.

	Min	Max	Moyenne
À partir de G1	1993	4360	3140.03
À partir de HEN1	0	5999	956.96

TABLE 5.1 – Débit généré par les drones en kbits/sec

Les résultats montrent que le débit envoyé par le drone G1 est supérieur à celui envoyé par le drone HEN1. Cela est dû au fait qu'un filtre supplémentaire d'estimation d'altitude est activé sur G1 ce qui causent la génération de plus de données.

Les courbes de la figure 5.10b montrent l'effet de la perte de liaison directe entre HEN1 et la station sol, traduit par la différence entre le débit émis par le drone et le débit reçu par la station sol qui commence avec le passage de HEN1 derrière le bâtiment.

Taux de perte
Le taux de perte est calculé pour les périodes avec et sans routage par la comparaison des captures de trafic réalisées par `tcpdump`. Le tableau 5.2 présente ces résultats.

Source	Destination	Taux de perte		
		Sans routage	Avec routage	Totale
Station sol	G1	0.13	3.73	1.03
Station sol	HEN1	0.38	43.44	9.72
G1	Station sol	0	0.71	0.15
HEN1	Station sol	1.68	23.55	7.04

TABLE 5.2 – Taux de perte (%) mesurés durant les différentes périodes de l'expérimentation

Les pertes sont concentrés au début de la phase avec routage et après sa fin. En effet, des paquets sont envoyés directement de la station sol vers le drone HEN1 alors qu'il est situé hors de sa portée. Cette erreur est causée par AODV qui ne détecte pas les pertes de connexion immédiatement et peut avoir des situations d'instabilité au niveau de sa table de routage.
Toutes ces pertes sont dues à des problèmes de routage puisque aucun paquet n'a été perdu par congestion au niveau G1.

Cette expérimentation nous a permis de valider le fonctionnement du nouveau système Paparazzi avec toutes les modifications matérielles et logicielle, ainsi que son contrôle à travers un réseau ad hoc.

Ce système permet de passer de l'architecture de communication en étoile utilisé par Paparazzi vers une structure en maille permettant de transmettre les messages vers les drones identifiés par leurs adresses IP.

5.3.2 Évaluation de DAN

À cause de la perte de l'un des quadricoptères modifiés suite à un accident, nous avons été obligé de réaliser nos expérimentations avec un seul drone, l'autre a été remplacé par une carte *Raspberry Pi*. Cette carte échange un vrai trafic Paparazzi produit par un générateur de trafic que nous avons créé spécialement pour ces expérimentations. Il génère des paquets conformes aux vrais messages Paparazzi en se basant sur les données enregistrées dans les fichiers *log* des anciens vols réels.

Le but de cette expérimentation est d'étudier et de valider le fonctionnement de DAN dans un environnement réel. Pour cette raison, quatre scénarios

ont été définis permettant chacun d'étudier un cas de fonctionnement spécifique :

- **1er scénario** : réalisation du même scénario de validation de fonctionnement du contrôleur d'admission évalué par simulation dans la section 4.2.1 ;
- **2eme scénario** : une évaluation du fonctionnement de DAN dans le cas d'une saturation totale du canal de transmission. Ces résultats sont comparés aux résultats du même scénario obtenu sans DAN ;
- **3eme scénario** : une évaluation de DAN dans le cadre des échanges Paparazzi. Durant cette expérimentation, les nœuds échangent des vidéos en temps réel en plus du trafic Paparazzi. Le trafic est envoyé dans un seul sens ; des nœuds vers la station sol ;
- **4ème scénario** : réalisation du scénario 3 avec des échanges entre les nœuds et la station de contrôle d'une part et entre les nœuds d'autres part.

Pour les scénarios 2, 3 et 4 nous avons essayé de garder le même principe que l'expérimentation précédente (figure 5.11). Deux nœuds s'éloignent simultanément de la station sol. Par la suite, l'un d'eux s'arrête alors que l'autre continue son chemin et perd sa liaison directe avec la station sol. Après une courte période dans cet état, les nœuds reviennent ensemble à leurs positions initiales.

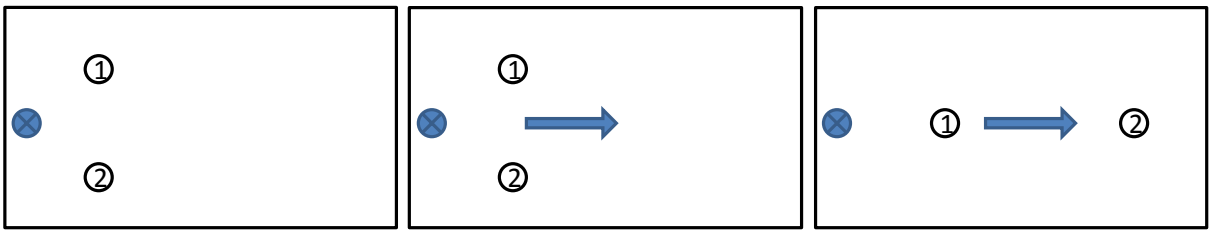


FIGURE 5.11 – Scénario des expérimentations

5.3.2.1 Validation du fonctionnement du contrôleur d'admission

Cette expérimentation a pour but la validation des mécanismes de DAN pour la gestion des demandes Premium dépassant sa capacité. Pour cela, le scénario simulé dans la section 4.2.1 est réutilisé ; deux nœuds fixes sont situés à la portée l'un de l'autre (un émetteur et un récepteur).
La source a trois flux Premium à envoyer qui dépassent sa capacité totale :

- Premium 1 : 10 Mbits/s ;

– Premium 2 : 5 Mbits/s ;
– Premium 3 : 15 Mbits/s.

La capacité totale du Premium dans le réseau est fixée à 20 Mbits/s.

L'admission est réalisée selon l'ordre chronologique de réception des requêtes de réservation de ressources.

Théoriquement, le CA peut accepter uniquement deux trafics (soit le 1 et le 2, soit le 2 et le 3). La troisième application sera notifiée dans le but de pouvoir initialiser une procédure de négociation avec le réseau.

Uniquement pour cette expérimentation, nous considérons que le flux non admis par le contrôleur d'admission continue à émettre son trafic sans dégradation de qualité, dans le but d'étudier la réaction de DAN face à ce trafic Premium non admis par le contrôleur d'admission.

Taux de Perte

Le tableau 5.3 présente les taux de perte mesurés pour les différentes classes de trafic.

	Flux 1 et Flux2	Flux 3
Pertes dues à la congestion	0	56.12
Pertes totales	0.5	60.32

TABLE 5.3 – Taux de perte mesuré pour les différents flux de trafic (%)

Les flux 1 et 2 traités comme flux Premium ont perdu 0,5% de leurs paquets alors que le flux 3 qui n'a pas réussi a réserver les ressources nécessaires, a été considéré comme un trafic Best-effort et il a perdu 60,32% de la totalité de ses paquets.

Ces pertes sont dues aux plusieurs facteurs comme les collisions, la congestion ou même des problèmes liés aux matériels. Les pertes par congestion du système sont causées par le rejet des paquets suite à la saturation de la file d'attente appropriée à cette classe de trafic. La différence entre les trois flux est remarquable (0% pour les flux 1 et 2 et 56,12% pour le flux 3).

Nous remarquons que le flux 3 a été traité comme du trafic Best-effort puisqu'il n'a pas été admis par le contrôleur d'admission.

Délai de bout en bout

Le délai de bout en bout est défini par le temps de transfert d'un paquet de bout en bout (entre l'émetteur et le récepteur). Il tient compte du temps de propagation du paquet le long du chemin vers sa destination ainsi que du temps de traitement au niveau des deux extrémités et de chaque nœud intermédiaire (comme le temps passé dans les files d'attente, etc).

Le tableau 5.4 présente les délais mesurés pour les flux échangés dans le réseau.

	Flux 1 et Flux 2	Flux 3
Délai de bout en bout (ms)	1.346	71.409

TABLE 5.4 – Délais de bout en bout mesurés pour les flux envoyés (ms)

Nous remarquons une grande différence entre le délai mesuré pour le flux 1 et 2 (1,346 ms) et celui mesuré pour le flux 3 (71,409 ms). En effet, l'émission des paquets du flux 3, considérés comme du trafic Best-effort, est retardée afin de respecter la priorité accordée aux paquets Premium.

Débit de transmission

La figure 5.12 présente le débit mesuré pour chaque flux de trafic. Ces valeurs sont mesurées chaque seconde.

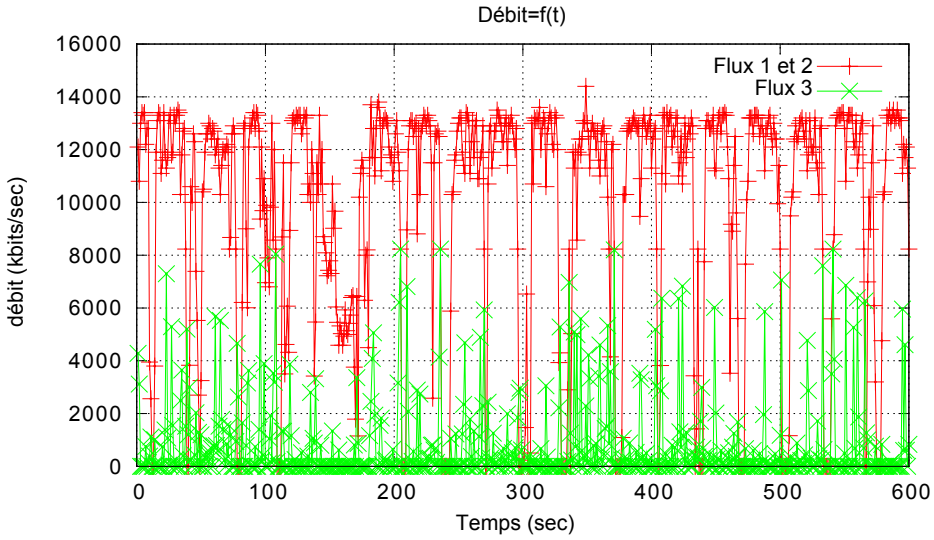


FIGURE 5.12 – Débits mesurés pour les différents flux de trafic

Les courbes montrent une différence remarquable entre le débit des flux 1 et 2 et le débit de flux 3 et confirme le fait que le flux 3 est considéré comme un trafic Best-effort, contrairement aux flux 1 et 2 qui bénéficient des débits demandés.

Cette expérimentation montre la capacité de DAN à contrôler les émissions de flux Premium, à respecter les priorités entre les classes de trafic et à garantir les ressources nécessaires pour les flux Premium admis.

5.3.2.2 Test de saturation de canal

Les mécanismes de gestion de QdS doivent gérer le cas de congestion du réseau tout en garantissant les priorités et la QdS demandée pour chaque classe de trafic. Cette expérimentation a pour but d'étudier le fonctionnement de DAN dans un cas de congestion du réseau afin de valider ses mécanismes.

Pour saturer le canal, nous avons utilisé le même scénario décrit précédemment, deux nœuds envoient du trafic vers une seule station sol. Chaque nœud émet un trafic constitué de :

- Urgent : 7 kbits/s ;
- Premium : 5 Mbits/s ;
- Best-effort : 15 Mbits/s.

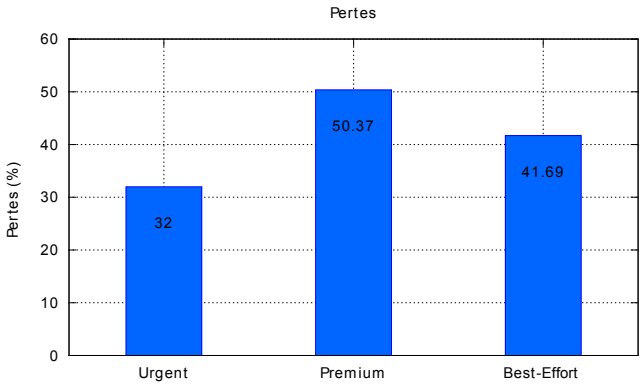
La capacité totale du Premium est fixée à 10 Mbits/s.

Ce scénario est réalisé deux fois ; une première fois avec les mécanismes de gestion de QdS de DAN et une deuxième avec le système de communication ad hoc ordinaire (sans les mécanismes QdS de DAN) dans le but de pouvoir comparer les résultats obtenus qui sont présentés dans les paragraphes suivants.

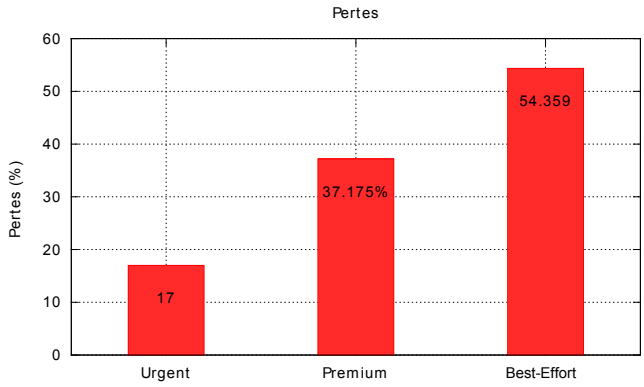
Taux de perte

Les figures 5.13 présentent le taux de perte par classe de trafic mesurés pour chaque système (avec et sans DAN). La première constations concernant ce graphe 5.13b est l'ordonnancement des barres de l'histogramme. Le taux de perte le plus faible est accordé à la classe Urgent alors que la classe Best-effort présente le plus de pertes. Cela montre que DAN respecte les priorités entre les différentes classes de trafic.

De plus nous remarquons, une amélioration au niveau du taux des pertes du trafic Premium qui passe de 50,37% à 37,175%. Cette amélioration est due au fait que la classe Premium est plus prioritaire que la classe Best-effort. En effet, avec DAN, les paquets Best-effort laissent la priorité d'émission aux paquets Premium. Par conséquent, ils se cumulent dans la file d'attente en attendant leur tour pour être émis et causent la saturation. Chaque file d'attente a une capacité de stockage limitée. Une fois saturée, elle commence à jeter les nouveaux paquets Best-effort arrivant. En conséquence, le nombre de paquets Best-effort détruits par la file d'attente à cause de la congestion augmente (passe de 41,60% à 54,350%) alors que ceux de la classe Premium et la classe Urgent diminuent.



Sans DAN
(a)



Avec DAN
(b)

FIGURE 5.13 – Taux de perte mesurés pour chaque classe de trafic avec et sans DAN

En plus de la congestion, ces pertes sont causées par d'autres phénomènes comme la perte de connectivité et les collisions dues aux interférences. La figure 5.5 présente le taux de perte causées uniquement par congestion. Ces pertes sont mesurées au niveau de chaque file d'attente :

$$\text{Taux de perte par congestion} = \frac{Nombre\,de\,paquets\,détruits}{Nombre\,de\,paquets\,total}$$

	Urgent	Premium	Best-effort
Pertes dues à la congestion	0	0	31.928

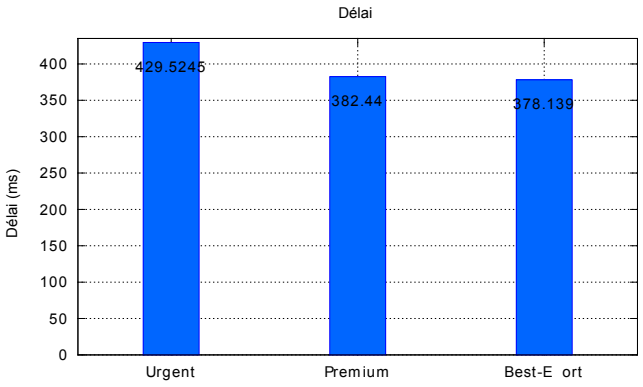
TABLE 5.5 – Taux de perte dues à la congestion du réseau

Aucun paquet des deux classes Urgent et Premium n'a été détruit par congestion.

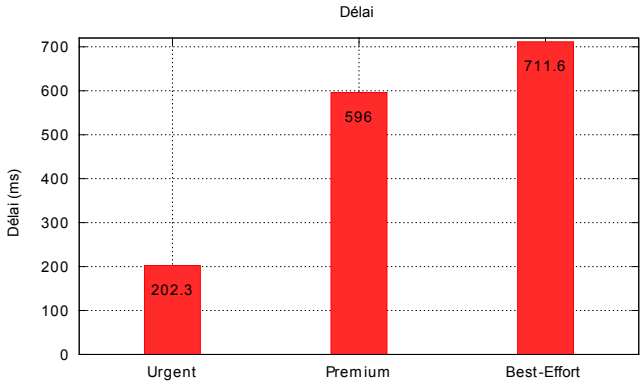
Délai de bout en bout

Le délai est calculé en se basant sur les fichiers d'enregistrement de trafic. Nous calculons l'intervalle de temps entre le moment de réception (au niveau du récepteur final) et le moment d'émission (au niveau du nœud source).

La figure 5.14 présente les délais de bout en bout calculés pour chaque classe de trafic durant les deux expérimentations réalisées (sans et avec DAN).



Sans DAN
(a)



Avec DAN
(b)

FIGURE 5.14 – Délais mesurés pour chaque classe de trafic avec et sans DAN

Comme pour les taux de perte, la première chose que nous remarquons est l'organisation du format des barres de l'histogramme 5.14b. Il est clair que la classe *Urgent* a le délai le plus faible (202,3 ms) alors que la classe Best-effort a le délai le plus important (711,6 ms). En effet, pour respecter la priorité entre les différentes classes de trafic, le système retarde l'émission des paquets Best-effort. Ces paquets doivent attendre leurs tours d'émission jusqu'à ce que les files Premium et Urgent soient vides ce qui explique l'écart entre les délais des différentes classes de trafic.

Sans DAN, les paquets sont traités de la même manière, indépendamment de leurs classes de trafic. Par conséquent, il n'y a pas de différence entre les délais mesurés. Par contre, sur la figure 5.14a, nous remarquons que les messages de la classe Urgent ont un délai plus élevé que les autres (un délai de 429,5245 ms pour les paquets Urgent et un délai de l'ordre de 380 ms pour les deux autres classes). Cela s'explique par le fait que les paquets Urgent sont *un phénomène rare* puisque le nombre de paquets Urgent envoyés est beaucoup moins important que celui des classes Premium et Best-effort ce qui pénalise le calcul des moyennes.

La saturation du canal a aussi un effet sur l'augmentation des valeurs du délai puisque chaque nœud doit attendre la libération du canal pour pouvoir émettre son trafic.

Débit de transmission

Les courbes 5.15 et 5.16 présentent les valeurs de débit mesurées durant les deux expérimentations (sans et avec DAN) pour les classes Premium et Best-effort. Les valeurs présentées sont les moyennes calculées sur des intervalles de 10 secondes pour des valeurs de débit mesurées chaque seconde.

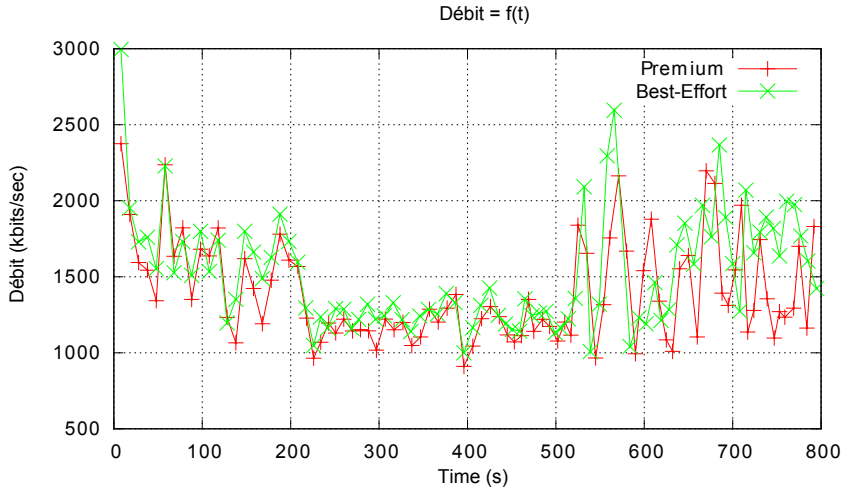


FIGURE 5.15 – Débit mesuré durant l'expérimentation sans DAN pour les classes Premium et Best-effort

Le tableau 5.6 présente les valeurs minimales, maximales ainsi que les moyennes de valeurs de débit mesurées.

Il est clair que le débit de la classe Premium est fortement amélioré grâce aux mécanismes de gestion de QdS. En effet, la valeur moyenne de débit de la classe Premium est passé de 1,374 Mbits/s (avec le système sans DAN) à

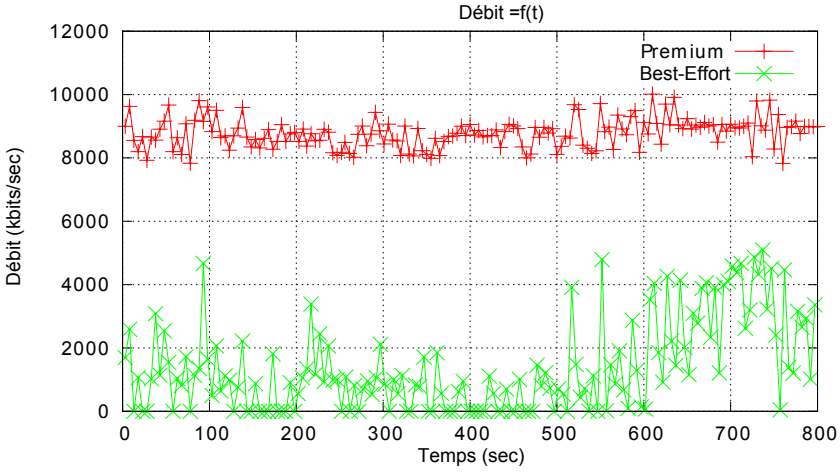


FIGURE 5.16 – Débit mesuré durant l'expérimentation avec DAN pour les classes Premium et Best-effort

8,769 Mbits/s (avec DAN). La forme de la courbe de débit mesuré pour la classe Premium dans la figure 5.16 montre que DAN a réussi à garantir le débit quasi-stable pour ces flux.

		Min	Max	Moyenne
Sans DAN	Premium	910,31	2236,1	1374,75
	Best-effort	999,95	2595,08	1495,86
Avec DAN	Premium	7827,64	10015,42	8769,07
	Best-effort	0	5093,06	1376,44

TABLE 5.6 – Débit mesuré pour les classes Premium et Best-effort durant les deux expérimentations (kbits/s)

Cette expérimentation permet de tester l'architecture de DAN dans un cas extrême de saturation du canal. Les résultats obtenus montrent que DAN est capable de gérer la QdS dans un réseau congestionné. Il classifie le trafic selon la classe demandée et respecte les différentes demandes de chaque catégorie de trafic. En effet, avec DAN, le trafic de classe Urgent bénéficient de la plus haute priorité : du délai de bout en bout ainsi que du taux de perte les plus faibles. De plus, DAN respecte le débit demandée de la classe Premium.

5.3.2.3 Évaluation dans le contexte Paparazzi

Cette expérimentation a pour but d'étudier l'influence de DAN sur un réseau échangeant le trafic Paparazzi en plus d'un trafic de transmission vidéo de qualité HD (Haute Définition) en temps réel. Les deux nœuds utilisés dans cette expérimentation émettent leurs trafics dans un seul sens ; vers la station sol (liaison descendante).

Le trafic vidéo est de capacité 1.5 Mbits/s. Il est créé par un générateur de trafic CBR (*Constant Bit Rate*) émettant des paquets UDP de taille fixe (187,5 Octets chaque 1 ms).

L'expérimentation a duré 800 secondes et nous a permis de calculer le taux de perte, le délai ainsi que le débit de chaque classe de trafic.

Taux de perte

Les résultats montrent que ce trafic ne congestionne pas le réseau. En effet, le taux de perte causée par congestion au niveau des files d'attentes est égal à zéro. Dans ce cas, les pertes mesurées (présentées dans le tableau 5.7) sont la cause des problèmes de réseau, comme les collisions ou les pertes de connectivité entre les nœuds dues à une instabilité au niveau du protocole de routage.

	Urgent	Premium	Best-effort
Pertes dues à la congestion	0	0	0
Pertes totales	1.2	8.67	9.5

TABLE 5.7 – Taux de perte mesuré pour chaque classe de trafic (%)

Délai de bout en bout

Contrairement au taux de perte, le délai de bout en bout peut créer la différence dans ce cas. En effet, bien que ce trafic ne congestionne pas les files d'attente, il les oblige à retarder l'émission de certains paquets dans le but de respecter la priorité de chaque classe de trafic. Le tableau 5.8 présente les délais de bout en bout mesurés pour les différentes classes de trafic durant cette expérimentation.

Nous remarquons une différence entre le délai de la classe Best-effort (26 ms) et celui des deux classes Urgent et Premium (environ 2,7 ms pour chacune) suite au retardement de l'émission des paquets Best-effort par le système afin de respecter la priorité des autres classes.

	Urgent	Premium	Best-effort
Délai de bout en bout (ms)	2.702	2.729	26.479

TABLE 5.8 – Délai mesuré pour chaque classe de trafic

Débit de transmission

La figure 5.17 et le tableau 5.9 présentent le débit mesuré durant l'expérimentation pour les deux classes de trafic Premium et Urgent. Ces valeurs sont calculées de la même manière que pour les autres expérimentations. Nous mesurons le débit à chaque seconde. Par la suite, nous calculons la moyenne sur des intervalles de 5 secondes.

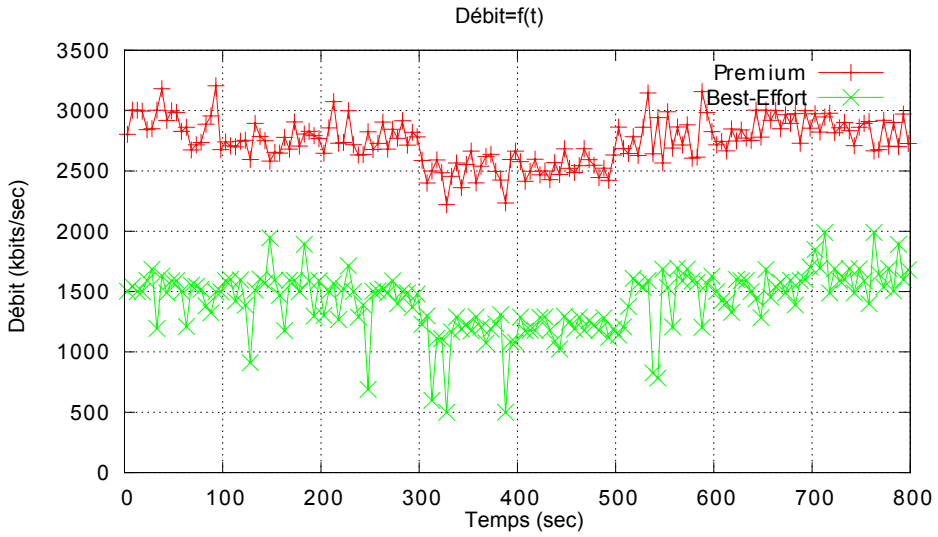


FIGURE 5.17 – Débits mesurés pour les classes Premium et Best-effort

	Min	Max	Moyenne
Premium	2220.8	3206	2742,93
Best-effort	500	1989,4	1417,041

TABLE 5.9 – Débits mesurés pour les deux classes Premium et Best-effort (kbits/s)

Les résultats montrent que le système a réussi à fournir le débit optimal pour les flux Premium.

Comme toutes les autres expérimentations, ce scénario a été réalisé plusieurs fois dans le but d'avoir différents cas possibles. Ce qui nous a permis de remarquer que les performances de DAN se dégradent dans une seule situation. Il s'agit du cas où le réseau rencontre des problèmes de connectivité.

La figure 5.18 présente le débit mesuré durant une expérimentation rencontrant ce problème. Elle montre une baisse de débit pour les deux flux entre 300 sec et 500 sec.

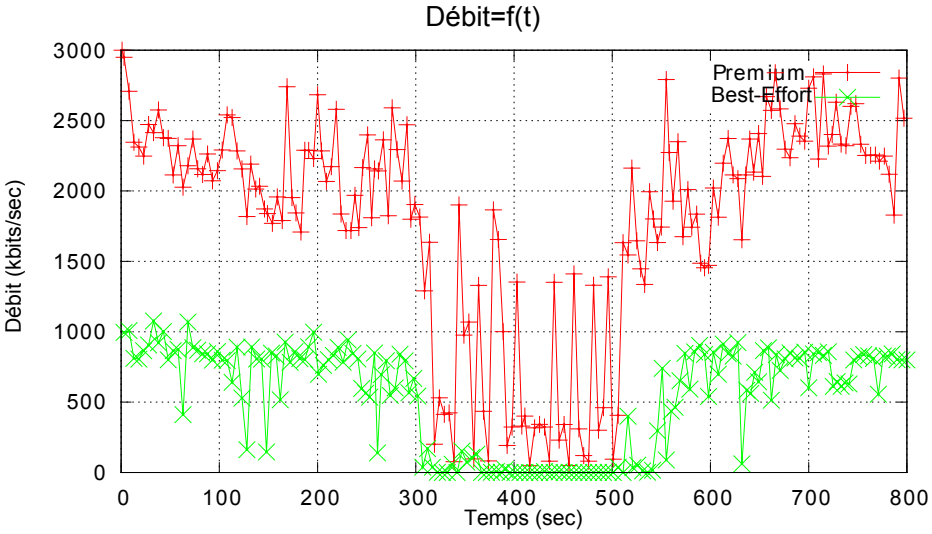


FIGURE 5.18 – Débit mesuré durant une expérimentation

Cette variation est due à des problèmes de connexion dans le réseau qui peuvent être causés par le protocole de routage ou par des facteurs de matériels. En effet, la figure 5.19, qui présente le nombre de paquets ICMP échangés dans le réseau, montre que durant cette période (300 sec et 500 sec), le nombre de paquets reçus est quasi-nul durant cet intervalle de temps. En conséquence, il est clair que le réseau a rencontré des problèmes de connectivité entre les nœuds durant cet intervalle de temps.

Cette expérimentation montre que DAN différencie le trafic échangé dans le système Paparazzi sans amélioration de la connectivité dans le réseau. En effet, il garantit la haute priorité (taux de perte et délai de bout en bout les plus faibles) pour la classe Urgent ainsi que le débit optimal pour la classe Premium dans la limite de la connectivité et des ressources disponibles entre

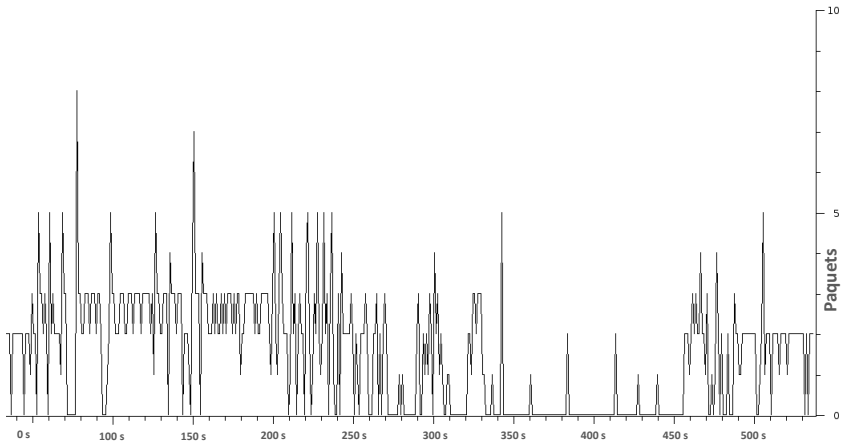


FIGURE 5.19 – Nombre de paquets ICMP échangés

les différents nœuds.

5.3.2.4 Évaluation dans le contexte Paparazzi avec des échanges dans différentes directions

Le but de cette expérimentation est d'étudier l'influence de DAN sur le système Paparazzi dans le cas où les nœuds échangent du trafic entre eux et avec la station sol. Ce cas d'échange crée plus de points de congestion dans le réseau.

Le trafic utilisé pour cette expérimentation est le même que celui utilisé pour l'expérimentation précédente : trafic Paparazzi en plus de flux vidéo HD en temps réel pour chaque émetteur.

Les trafics Urgent ainsi que le trafic Premium sont envoyés des nœuds vers la station sol. Le trafic Best-effort est émis entre les nœuds et entre les nœuds et la station sol.

Comme pour l'expérimentation précédente, ce trafic ne congestionne pas le réseau. Pour cette raison, nous n'allons pas présenter les taux de perte relevés.

Délai de bout en bout

Le tableau 5.10 présente le délai de bout en bout mesuré pour chaque classe de trafic.

Nous remarquons une grande différence entre le délai de la classe Urgent (2,4 ms) et celui de la classe Best-effort (30 ms). Cela est dû au fait que

	Urgent	Premium	Best-effort
Délai de bout en bout (ms)	2.4	3.55	30

TABLE 5.10 – Délai mesuré pour chaque classe de trafic (ms)

DAN retarde l'émission des paquets Best-effort pour privilégier les paquets des classes les plus prioritaires.

DAN respecte la différentiation de services et les exigences de chaque classe en termes de délai de bout en bout même dans ce cas de le trafic.

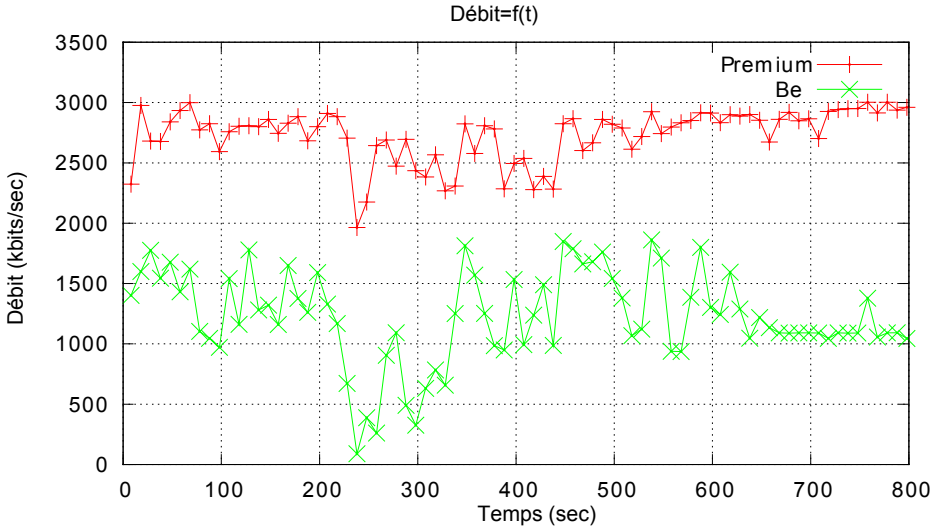


FIGURE 5.20 – Débits mesurés pour les deux classes Premium et Best-effort

Débit de transmission

La figure 5.20 présente les débits mesurés durant l'expérimentation des deux classes de trafic Premium et Best-effort. La courbe montre que le débit demandé pour la classe Premium a été respecté.

Les résultats de cette expérimentation montrent que DAN peut s'adapter aux différentes situations de trafic, tout en garantissant une différenciation entre les services utilisés.

DAN est un système indépendant des protocoles MAC et routage.

Conclusion

Ce chapitre a présenté les modifications logicielles et matérielles réalisées dans le but d'intégrer DAN au système Paparazzi. Cette intégration a abouti à un système de communication permettant de faciliter le contrôle d'une flotte de drones coopératifs par la station sol, en offrant le support nécessaire qui respecte les besoins de différents types de données échangés à travers un réseau ad hoc. Ce système différencie le trafic et garantie différents niveau de QdS.

Afin d'évaluer les performances de ce système dans son environnement réel, nous avons réalisé une variété d'expérimentations. Les résultats montrent que DAN répond aux exigences du système. Il différencie le service et garantie la QdS demandée pour chaque classe. En effet, il limite les pertes des deux classes Urgent et Premium grâce à son mécanisme d'ordonnancement qui élimine les pertes par congestion et garantit le délai de bout en bout le plus faible pour ces deux catégories de trafic. De plus, il respecte le débit demandé par les flux Premium grâce au mécanisme de réservation de ressources.

DAN est un système qui ne demande pas des modifications complexes au niveau de l'architecture de communication usuelle et qui répond au exigence du système de flotte de drones coopératifs.

Chapitre 6

Conclusions et perspectives

Contents

6.1 Travaux réalisés 143
6.1.1 L'architecture DAN 144
6.1.2 Intégration DAN au système Paparazzi 145
6.2 Perspectives 147
6.2.1 Perspectives réseaux 147
6.2.2 Opérations pour les drones 149

Ce chapitre résume les différentes contributions réalisées dans le cadre de cette thèse et présente des idées qui peuvent être traitées comme une continuité de ce travail.

6.1 Travaux réalisés

L'utilisation d'une flotte de drones dans des missions civiles coopératives nécessite un système de communication permettant de contrôler chaque aéronef et le connecter à la station de contrôle ainsi qu'aux autres drones afin de pouvoir échanger différents types de données. Les réseaux ad hoc sont une solution pour garantir un niveau élevé de coordination entre les différents agents de ce système.

Le but de cette thèse est de créer une nouvelle structure de communication permettant d'offrir au système Paparazzi un support conscient des besoins de chaque émetteur, permettant de faciliter le contrôle d'une flotte de drones au cours d'une mission coopérative. Cette structure est capable de répondre aux exigences du système en termes de QdS. En effet, elle doit différencie le

trafic selon les besoins de chaque drone qui peuvent être variables au cours du temps.

Dans ce genre de situations, les drones jouent différents rôles dans l'avancement de l'opération et échangent divers types de message. Par conséquent, ils ont des besoins spécifiques en termes de communication qui peuvent évoluer au cours du temps. Par exemple, les messages contenant des informations critiques sur l'état du système nécessitent un maximum de fiabilité. Ces messages doivent bénéficier du taux de perte et du délai de bout en bout les plus faibles.

À partir de l'étude de fichiers d'enregistrement de trafics échangés entre les drones Paparazzi et la station de contrôle, nous avons pu classifier les messages en trois catégories essentielles : une première classe *Urgent* pour les messages critiques qui nécessitent la priorité absolue, une deuxième classe *Premium* destinée aux flux de trafic ayant des demandes strictes en termes de QdS et une troisième classe *Best-effort* qui englobe le reste du trafic qui n'a aucune exigence de QdS.

Afin de valider le fonctionnement de ce système, il est nécessaire d'évaluer ses performances afin de prédire les problèmes qui peuvent l'affecter en environnement réel. Pour notre étude nous nous sommes basé sur deux moyens d'évaluations qui sont la simulation et l'expérimentation réelle.

6.1.1 L'architecture DAN

DAN est l'architecture de communication que nous proposons en réponse aux exigences présentées dans la section précédente. Elle groupe des mécanismes de gestion de QdS permettant de différentier le trafic et de garantir les besoins demandées pour chaque classe.

La spécificité de DAN est d'offrir à l'application la possibilité de gérer les besoins en QdS de ses flux grâce à son API permettant d'échanger des informations concernant la capacité du réseau. Par conséquent, c'est à l'application seule de décider d'annuler son émission ou de dégrader sa qualité et de négocier avec le réseau.

DAN permet au niveau applicatif d'exprimer les besoins de chaque flux en termes de QdS. Ainsi, Il reçoit des renseignements sur chaque flux à émettre pour pouvoir le classer et marquer ses paquets selon sa catégorie de trafic.

Par la suite, un algorithme d'ordonnancement est utilisé permettant de retarder l'émission des classes les moins prioritaires pour respecter la priorité des autres classes.
Pour la catégorie Premium, DAN offre une garantie stricte sur la QdS grâce à son mécanisme de réservation de ressources. Le but de limiter l'utilisation

de la signalisation uniquement pour la classe Premium est d'économiser la capacité du réseau.

Afin de respecter la QdS sans pénaliser la classe Best-effort qui représente plus de 60% du trafic échangé, nous avons fixé la capacité totale du trafic Premium dans le système. Plus précisément, il s'agit d'une limitation du maximum de bande passante utilisable pour les flux Premium (cette valeur est égale à 20% de la capacité totale du canal de transmission). DAN est capable de garantir la QdS demandée pour chaque flux dans la limite de cette capacité.
Bien que dans d'autres situations, cette méthode puisse être la cause d'un gaspillage de bande passante puisqu'elle oblige les flux à respecter leur limite de débit, même s'il n'y a pas de congestion, elle est adaptée à notre système qui échange plus de trafic Best-effort que de Premium car elle permet de ne pas dégrader les autres flux.

Afin de créer un environnement proche de la réalité permettant d'étudier les performances de DAN et de prédire les problèmes qui peuvent affecter le système dans un environnement réel, nous avons conçu un modèle de mobilité reproduisant les vrais mouvements d'un drone Paparazzi, PPRZM.
PPRZM met en œuvre des types de mouvement qui ne sont pas pris en compte par les autres modèles de mobilité ordinaires (comme la répétition d'un même mouvement plusieurs fois dans la même zone ou les mouvements d'aller-retour, etc). Nous avons choisi d'utiliser ce modèle de mobilité puisque dans le cas d'une flotte de drones, l'obtention des traces de plusieurs aéronefs est difficile à cause de la complexité de lancement de plusieurs vols. PPRZM offre la possibilité de faire varier les paramètres de simulation comme la durée du scénario ou le nombre de drones utilisés dans le but d'étudier une diversité de cas possibles. Ce qui nous a permis de créer des scénarios de simulation difficiles à réaliser avec des drones réels ç cause, par exemple, du nombre élevé d'aéronefs utilisés.

6.1.2 Intégration DAN au système Paparazzi

Une étape importante de cette thèse, est l'intégration de DAN au système Paparazzi. Cette intégration permettra au système Paparazzi de passer de sa méthode de contrôle classique en étoile à travers un réseau 802.15.4 supportant des émissions point à multi-points (sur la liaison montante) et point à point (sur la liaison descendante), vers une autre méthode utilisant un réseau maillé offrant la possibilité d'adresser chaque nœud par une adresse IP.

Pour la réalisation, des modifications matérielles et logicielles ont été nécessaires afin de pouvoir implémenter l'architecture DAN et de l'intégrer au système de communication de Paparazzi.

Ce nouveau système de communication a été étudié par des expérimentations réelles dans plusieurs situations. Les résultats de ces études montrent que DAN est capable de répondre aux exigences du système. En effet, il différentie le service et garantit la QdS demandée pour chaque classe de trafic. Pour les messages de la classe Urgent, il offre la priorité absolue se traduisant par un taux de perte et un délai de bout en bout les plus faibles et garantit le débit demandé pour chaque flux de trafic Premium. Pour ces deux classes, DAN réduit au maximum les pertes puisque il élimine les pertes dues à la congestion du système.

En outre, les résultats montrent que même dans les réseaux les plus congestionnés, DAN gère la situation sans trop pénaliser le trafic Best-effort.

Après avoir réalisé notre étude avec deux moyens différents : la simulation et les expérimentations réelles, nous pouvons comparer les deux, loin des définitions et des comparaisons théoriques explicitées dans la littérature. Nous avons rencontré quatre différences majeurs :

- **Implémentation du système** : l'implémentation du nouveau module ou protocole au niveau d'un simulateur réseau est plus simple qu'une implémentation réelle puisqu'elle est plutôt guidée, en utilisant des modules virtuels (des classes) imitant les fonctionnalités de l'architecture de communication (IP, applications, UDP, ARP, etc.) alors que pour l'implémentation réelle, il fallait traiter différents processus et choisir les librairies qui aident à manipuler l'architecture Linux ;
- **Les scénarios** : les scénarios de simulation sont caractérisés par leur simplicité de réalisation et leur flexibilité. En effet, ils supportent la définition de situations délicates avec un nombre de drones élevé, plusieurs répétitions, longues durées de simulation, etc. Au contraire, les scénarios d'expérimentation réelles sont moins flexibles notamment pour le nombre de mobiles, les répétitions et le lancement qui nécessitent la présence de plusieurs personnes et des facteurs métrologiques convenables (pas de pluie et pas de vents forts) ;
- **Contrôle des résultats** : avec le simulateur, l'enregistrement de l'historique du système est réalisé d'une manière automatique sans aucun coût pour le système étudié. Par contre, ces données peuvent couter cher pour les expérimentations réelles en termes de mémoire et de bande passante (si elles sont collectées en temps réel). Dans ce cas, il faut optimiser les données à enregistrer au maximum pour ne pas affecter les performances du système ;
- **Facteurs affectant les résultats** : plusieurs facteurs ne sont pas pris en compte par le simulateur réseau pourtant ils ont une influence sur les performances du système comme la capacité du canal de transmission qui se dégrade énormément dans le cas réel alors que pour le simu-

lateur elle garde sa valeur théorique, les batteries limitées, les pannes matérielles ou logicielles et même les échecs et l'instabilité du protocole de routage qui peuvent causer des pertes de connectivité dans le vrai système. En outre, nous avons remarqué la sensibilité du système sans fil réel face aux interférences et aux obstacles ainsi que l'impact de la distance entre les nœuds sur la qualité des liens entre eux.

Toutes ces différences entre la simulation et les expérimentations réelles montrent que ces deux moyens sont complémentaires. En effet, la simulation facilite la réalisation d'études de scénarios délicats alors que l'expérimentation réelle met le système dans son environnement d'application réel entouré de tous les obstacles qui peuvent affecter ses performances.

La réalisation de l'architecture DAN a abouti à une plate-forme de test qui peut être utilisée pour la réalisation d'autres expérimentations pour des protocoles de communication conçus pour les drones.

6.2 Perspectives

Si cette thèse apporte des améliorations dans le cadre des communications pour les drones léger civils, plusieurs pistes de travaux futurs se dessinent.

6.2.1 Perspectives réseaux

Cette section présente des idées et des améliorations possibles pour renforcer les travaux réalisés dans le domaine de la recherche réseau.

Protocole de routage muti-chemin

Il serait intéressant d'associer l'architecture DAN avec un protocole de routage multi-chemin, qui permet d'établir plusieurs routes entre chaque source et destination. AOMDV est un exemple de ces protocoles. Il permet d'établir des routes disjointes qui ne partagent aucun nœud intermédiaire entre chaque source et sa destination. AOMDV permet d'améliorer les performances de AODV en termes de délai et échanges de messages de contrôle dans le réseau.

Avec un grand nombre de nœuds, cette proposition offre à DAN la possibilité de répartir la charge dans le réseau selon la classe et les besoins du trafic en termes de QdS. En effet, dans le cas d'échec de réservation de ressources pour un flux Premium, le système peut tenter directement une

nouvelle réservation en passant par une route alternative. En conséquence, entre chaque nœud source et sa destination, les paquets Premium peuvent traverser différentes routes.

Cette amélioration permettra de répartir la charge dans de grands réseaux ce qui améliore les performances du système en termes de délai, de pertes et de débit pour les deux classes de trafic Premium et Best-effort.

QdS et interférences

Pour mieux adapter DAN aux grands réseaux dynamiques, il pourrait être complété par un mécanisme permettant d'estimer la bande passante disponible localement en tenant compte des interférence causées par les voisins. Afn de respecter toutes les exigences de conception de l'architecture DAN, ce mécanisme doit être indépendant du niveau MAC et doit consommer le minimum de ressources possibles. Deux possibilité peuvent être utilisées ainsi :

- Émission de signalisation spécifique périodiquement par le contrôleur d'admission pour sonder la qualité du canal ;
- Utilisation des messages de contrôle du protocole de routage comme indication sur la qualité du canal ;
- Utilisation des mesures passives sur le trafic reçu.

Ces mécanismes permettent d'augmenter la certitude de l'estimation du débit disponible localement au niveau de chaque nœud ce qui améliore les performances du système notamment dans les réseaux très dynamique.

Adaptation de la durée du minuteur utilisé par DAN avec d'autres application

Dans le but d'adapter DAN à d'autres scénarios d'utilisation, la durée de l'intervalle *WTC* pourrait varier avec chaque flux de trafic Premium. En effet, elle pourrait prendre en compte l'intervalle de temps qui sépare l'émission de deux paquets de données consécutifs ou encore le temps de traitement au niveau de chaque nœud intermédiaire puisque cette durée varie selon la capacité du système utilisé. Cette amélioration permettrait de fixer une durée maximale d'attente en fonction des caractéristiques de chaque flux et la capacité des nœuds dans le réseau.

Simulations et modèle de mobilité

Afin de garantir un environnement de simulation pour les réseaux ad hoc de drones, plus proche de la réalité, le modèle de mobilité PPRZM pourrait être complété par des modifications. Tout d'abord, il pourrait générer des mouvements pour un essaim de drones (*swarm*). Plus précisément, les drones

auront la possibilité de bouger en coordination totale : de réaliser le même mouvement, au même moment mais chacun dans sa position. De plus, il est possible de compléter PPRZM aussi par d'autres mouvements non supportés par les drones Paparazzi qui impliquent une variation de l'altitude (comme les rotations en spirale verticale en montant ou descendant).

6.2.2 Opérations pour les drones

Cette thèse apporte des nouveautés dans le cadre des communications de drones civils ce qui permet de créer de nouvelles opérations et de développer de nouvelles capacités complexes puisqu'elle offre une méthode plus simple pour contrôler les aéronefs en garantissant les besoins en termes de QdS pour chaque service.

Tout d'abord, ce travail a abouti à un nouveau système de communication pour l'auto-pilote Paparazzi qui peut être mis en disposition de toute la communauté qui utilise ce système libre. En conséquence, il permettra de réaliser les missions conçues pour les flottes de drones et nécessitant ce moyen de contrôle des drones.

En outre, il serait intéressant d'optimiser les fonctionnalités du système Paparazzi qui étaient utilisées auparavant avec le système de contrôle en étoile comme par exemple l'évitement de collision puisque les drones ne communiquaient pas ensemble directement, c'est la station de contrôle qui détectait la possibilité d'avoir une collision en recevant périodiquement les positions des aéronefs, et par la suite, elle informait les aéronefs concernés pour qu'ils l'évitent.

De plus, ce nouveau système pourrait être utilisé pour tester de nouvelles fonctionnalités qui permettront d'utiliser les drones dans de nouvelles missions plus performantes. Ces fonctionnalités pourront changer leurs modes de fonctionnement selon la disponibilité des ressources dans le réseau puisqu'elles peuvent être à jour de la capacité du réseau grâce à ses échanges à travers l'API DAN. Par exemple, le drone peut modifier la résolution de sa vidéo émise en fonction de la capacité du réseau.

Publications

Conférences internationales avec comité de lecture

- Ouns BOUACHIR, Fabien GARCIA, Nicolas LARRIEU, Thierry GAYRAUD. **Ad hoc Network QoS Architecture For Cooperative Unmanned Aerial Vehicles (UAVs)**. Wireless Days (WD), 2013 IFIP , vol., no., pp.1,4, 13-15 Nov. Valencia, Spain 2013.
- Ouns Bouachir, Alinoe Abrassart, Fabien Garcia, Nicolas Larrieu. **A Mobility Model For UAV Ad hoc Network**. 2014 International Conference on Unmanned Aircraft Systems (ICUAS) May 27-30, 2014. Orlando, FL, USA

Conférences nationales

- Ouns BOUACHIR, Fabien GARCIA, Nicolas LARRIEU, Thierry GAYRAUD. **Conception et Implémentation d'une Architecture de Communication pour Agents Mobiles Coopératifs**. 14ème Congrès des Doctorants EDSYS (École Doctorale Systèmes), Tarbes, France, 2013

Rapports techniques

- 2 délivrables pour le projet D3CoS.

Acronymes

AANET Aeronautical Ad hoc Network

ACK Acquittement

AG Assured Forwarding

AMUAV Adaptive MAC protocol for UAV

AODV Ad hoc On-Demand Distance Vector

AOMDV Ad hoc Multi-path Distance Vector

API Application Programming Interface

PBQMRP Position Based QoS Multi-cast Routing Protocol

CA Contrôleur d'admission

CEDAR Core-Extraction Distributed Ad hoc Routing

CO Consigne de l'Opérateur

CSI Channel State Information

CSMA/CA Carrier Sense Multiple Access avec Collision Avoidance

CTS Clear To Send

D3CoS Designing Dynamic Distributed Cooperative Human-Machine Systems

DAN DCoS Ad hoc Network

DCF Distributed Coordination Function

DCoS Distributed Cooperative Human-Machine Systems

DiffServ Differentiated Services

DOLSR Directional Optimized Link State Routing

DSDV Destination-Sequenced Distance Vector

DSR Dynamic Source Routing

DTN Delay Tolerant Network

ECN Explicit Congestion Notification

EDCA Enhanced Distributed Channel Access

EF Expeditd Forwarding

EHSR Extended Hierarchical State Routing

FIFO First In First Out

FQMM Flexible QoS Model for MANETs

GCS Ground Control Station

GPMOR Geographic Position Mobility Oriented Routing

GPRS General Packet Radio Service

GPS Global Positioning System

GPSR Greedy Perimeter Stateless Routing

GSM Global System for Mobile Communications

HAP High altitude platform

HiperLAN High Performance radio LAN

IETF Internet Engineering Task Force

INSIGNIA In-Band Signaling Support for QoS In Mobile Ad hoc Networks

IntServ Integrated Services

IO Informations de l'Opération

IP Internet Protocol

IS Informations du Système

LAROD Location Aware Routing for Opportunistic Delay Tolerant

Libnl Netlink Protocol Library

LODMAC Location Oriented Directional MAC protocol

LTE Long Term Evolution

MAC Media Access Control

MAR Mobile Agent Routing

MANET Mobile Ad hoc Network

MC Messages de Contrôle

MC-TRACE Multicasting Time Reservation using Adaptive Control for Energy Efficiency

MH-TRACE Multi-Hop Time Reservation using Adaptive Control for Energy Efficiency

MPLS Multi-Protocol Label Switching

MPR Multi-packet reception

MTT Methods, Techniques and tools
OMNET++ Objective Modular Network Testbed in C++
OSLR Optimized Link State Routing Protocol
PCF Point Coordination Function
PHB Per Hop Behavior
PPRZM Paparazzi Mobility
PSTN Public Switched Telephone Betwork
QdS Qualité de Service
RD Random Direction
RGR Reactive-Greedy-Reactive routing protocol
RREQ Route Request
RREP Route Reply
RERR Route Error
RSVP Resources Setup Reservation Protocol
RTP Real-time Transport Protocol
RTS Request To Send
RWP Random Way-Point
SCTP Stream Control Transmission Protocol
SD Safe Distance
SRCM Semi-Random Circular Movement
SWAN Stateless Wireless Ad hoc Network
TARF Task Agent Resource Function
TCP Transmission Control Protocol
TDMA Time DivisionMultiple Access
TG Tête de groupe
ToS Type os Service
TTL Time To Live
UAANET UAV Ad hoc Network
UAV Unmanned Aerial System
UAV Unmanned Aerial Vehicle
UDP User Datagrame Protocol
UMTS Universal Mobile Telecommunications System

USMP UAV Search Mission Protocol

VANET Vehicular Ad hoc Network

WAVE Wireless Access in Vehicular Environment

WiMAX Worldwide Interoperability for Microwave Access

WPAN Wireless Personnal Area Network

WSN Wireless Sensor Network

WTC Wait To Clear

WTS Wait To Send

XTP Xpress Transfer Protocol

Bibliographie

[AD10a] A.I. Alshbatat and Liang Dong. Adaptive mac protocol for uav communication networks using directional antennas. In *International Conference on Networking, Sensing and Control (ICNSC), IEEE*, 2010.

[AD10b] A.I. Alshbatat and Liang Dong. Cross layer design for mobile ad-hoc unmanned aerial vehicle communication networks. In *International Conference on Networking, Sensing and Control (ICNSC), IEEE*, 2010.

[AHAN05] A. Al Hanbali, E. Altman, and P. Nain. A survey of tcp over ad hoc networks. *Communications Surveys Tutorials, IEEE*, 7(3) :22–36, Third 2005.

[BBC+98] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An architecture for differentiated services. IETF RFC 2475, December 1998.

[BCS94] R. Braden, D. Clark, and S. Shenker. Integrated services in the internet architecture. RFC 1633, Jun. 1994.

[BDJH04] Timothy X Brown, Sheetalkumar Doshi, Sushant Jadhav, and Jesse Himmelstein. Test bed for a wireless network on small uavs. pages 20–23, 2004.

[BEB+97] R. Braden, L. Zhang Ed., S. Berson, S. Herzog, and S. Jamin. Resource reservation protocol (rsvp). IETF RFC 2205, September 1997.

[Bes13] Frederic Besse. *Reseaux ad hoc aeronautiques.* PhD thesis, Ecole Nationale de l'Aviation Civile (ENAC), 2013.

[BGL08] Sunghyun Choi Byeong Gi Lee. *Broadband Wireless Access and Local Networks : Mobile WiMax and WiFi.* Artech House mobile communications series, 2008.

[BJ12] O. Bazan and M. Jaseemuddin. A survey on mac protocols for wireless adhoc networks with beamforming antennas. *Com-*

munications Surveys Tutorials, IEEE, 14(2) :216–239, Second 2012.

[Blo] John David Blom. *Unmanned Aerial Systems : A Historical Perspective*, chapter 1 and 2, pages 1–12 and 45–49. CreateSpace Independent Publishing Platform (September 1, 2010).

[Blu] Bluetooth. https ://www.bluetooth.org/en-us.

[BST13] Ilker Bekmezci, Ozgur Koray Sahingoz, and Samil Temel. Flying ad-hoc networks (fanets) : A survey. *Ad Hoc Networks*, 11(3) :1254–1270, 2013.

[BT11] P.-B. Bok and Y. Tuchelmann. Context-aware qos control for wireless mesh networks of uavs. In *Computer Communications and Networks (ICCCN), 2011 Proceedings of 20th International Conference on*, pages 1–6, July 2011.

[BVH07] Brett Bethke, Mario Valenti, and Jonathan How. Cooperative vision based estimation and tracking using multiple uavs. In *Conference of Cooperative Control and Optimization*, 2007.

[CBD02] Tracy Camp, Jeff Boleng, and Vanessa Davies. A survey of mobility models for ad hoc network research. In *Wireless Communications mobile Computing (WCMC) : Special Issue On Mobile Ad hoc Networking, Research, Trends and Applications*, volume 2, pages 483–502, 2002.

[CBM+05] D.W. Casbeer, R.W. Beard, T.W. McLain, Sai-Ming Li, and R.K. Mehra. Forest fire monitoring with multiple small uavs. In *American Control Conference, 2005. Proceedings of the 2005*, volume 5, 8-10, pages 3530– 3535, June 2005.

[CHKV06] Chen-Mou Cheng, Pai-Hsiang Hsiao, H. T. Kung, and D. Vlah. Performance measurement of 802.11a wireless links from uav to ground nodes with various antenna orientations. In *International Conference on Computer Communications and Networks, ICCCN*, 2006.

[CJS+10] Jung Il Choi, Mayank Jain, Kannan Srinivasan, Philip Levis, and Sachin Katti. Achieving single channel, full duplex wireless communication. In *ACM MobiCom'10 (Chicago, IL)*, 2010.

[CLM+11] S. Chaumette, R. Laplace, C. Mazel, R. Mirault, A. Dunand, Y. Lecoutre, and J-N Perbet. Carus, an operational retasking application for a swarm of autonomous uavs : First return on experience. In *Military Communications Conference, 2011- MILCOM*, 2011.

[CSGM13] DE Cook, PA Strong, SA Garrett, and RE Marshall. A small unmanned aerial system (uas) for coastal atmospheric research : preliminary results from new zealand. *Journal of the Royal Society of New Zealand*, 43(2) :108–115, 2013.

[CYCG07] James R. Clapper, JohnJ. Young, James E. Cartwright, and John G. Grimes. Unmanned systems roadmap 2007-2032. Technical report, Departement of Defense, USA, 2007.

[CYL+13] Yegui Cai, F.R. Yu, Jun Li, Yifeng Zhou, and L. Lamont. Medium access control for unmanned aerial vehicle (uav) adhoc networks with full-duplex radios and multipacket reception capability. *Vehicular Technology, IEEE Transactions on*, 62(1) :390–394, Jan 2013.

[dFB08] M. de Fatima Bento. Unmanned aerial vehicles. wo18.15 rking paper Inside GNSS, available on : http ://www.insidegnss.com/auto/janfeb08-wp.pdf, January/February 2008.

[ESB+04] D.C. Engelhart, Anand Sivasubramaniam, C.L. Barrett, M.V. Marathe, J.P. Smith, and M. Morin. A spatial analysis of mobility models : application to wireless ad hoc network simulation. In *Simulation Symposium, 2004. Proceedings. 37th Annual*, pages 35–42, April 2004.

[Ete08] K. Etemad. Overview of mobile wimax technology and evolution. *Communications Magazine, IEEE*, 46(10) :31–40, October 2008.

[FHS07] J. Hope Forsmann, Robert E. Hiromoto, and John Svoboda. A time-slotted ondemand routing protocol for mobile ad hoc unmanned vehicle systems. *Unmanned System Technology IX, SPIE*, 6561, 2007.

[GHWL13] Zhaoquan Gu, Qiang-Sheng Hua, Yuexuan Wang, and F.C.M. Lau. Reducing information gathering latency through mobile aerial sensor network. In *INFOCOM, 2013 Proceedings IEEE*, pages 656–664, April 2013.

[GsAS02] A. Veres Gahng-seop Ahn, Andrew T. Campbell and Li-H. Sun. Swan : Service differentiation in stateless wireless ad hoc networks. In *Proc. IEEE Infocom*, 2002.

[GSB09] S. Gowrishankar, S.K. Sarkar, and T. G. Basavaraju. Performance analysis of aodv, aodvuu, aomdv and raodv over ieee 802.15.4 in wireless sensor networks. In *Computer Science and*

Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on, pages 59–63, Aug 2009.

[HBM07] M.T. Hyland and M.A. Temple B.E. Mullins, R.O. Baldwin. Simulation-based performance evaluation of mobile ad hoc routing protocols in a swarm of unmanned aerial vehicles. In *21st International Conference on Advanced Information Networking and Applications Workshops, AINAW*, 2007.

[KK00] Brad Karp and H. T. Kung. Gpsr : Greedy perimeter stateless routing for wireless networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking, MobiCom*, 2000.

[KLHH05] T. Karhima, P. Lindroos, M. Hall, and S.-G. Haggman. A link level study of 802.11b mobile ad-hoc network in military environment. In *Military Communications Conference, MILCOM, IEEE*, 2005.

[KNt06] Erik Kuiper and Simin Nadjm-tehrani. Mobility models for uav group reconnaissance applications. In *in Proceedings of International Conference on Wireless and Mobile Communications. IEEE Computer Society*. IEEE, 2006.

[KNT08] E. Kuiper and S. Nadjm-Tehrani. Geographical routing in intermittently connected ad hoc networks. In *22nd International Conference on Advanced Information Networking and Applications - Workshops. AINAW*, 2008.

[Kru92] J. Kruys. Hiperlan, applications and requirements. In *Third IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 1992. Proceedings, PIMRC '92, pp.133-138, 19-21 Oct*, 1992.

[LC98] Seoung-Bum Lee and Andrew T. Campbell. Insignia : In-band signaling support for qos in mobile ad hoc networks. In *Workshop on Mobile Multimedia Communication MoMuc*, 1998.

[LCT+05] R. Levy, P.S. Carlos, A. Teittinen, L.S. Haynes, and C.J. Graff. Mobile agents routing-a survivable ad-hoc routing protocol. In *Military Communications Conference, 2005. MILCOM . IEEE*, 2005.

[lin] Linux advanced routing & traffic control. http ://www.lartc.org/.

[LMB09] R.L. Lidowski, B.E. Mullins, and R.O. Baldwin. A novel communications protocol using geographic routing for swarming

uavs performing a search mission. In *IEEE International Conference on Pervasive Computing and Communications, PerCom*, 2009.

[LPG06] M. Le, Joon-Sang Park, and M. Gerla. Uav assisted disruption tolerant routing. In *Military Communications Conference, IEEE MILCOM*, 2006.

[LRGM10] L. Lin, M. Roscheck, M. A. Goodrich, and B. S. Morse. Supporting wilderness search and rescue with integrated intelligence : Autonomy and information at the right time and the right place. In *Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI*, 2010.

[LSLY12] Lin LIN, Qibo SUN, Jinglin LI, and Fangchun YANG. A novel geographic position mobility oriented routing strategy for uavs. *Journal of Computational Information Systems*, 8 :709–716, 2012.

[LZH+06] Myung Jong Lee, Jianling Zheng, Xuhui Hu, Hsin hui Juan, Chunhui Zhu, Yong Liu, June Seung Yoon, and T.N. Saadawi. A new taxonomy of routing algorithms for wireless mobile ad hoc networks : the component approach. *Communications Magazine, IEEE*, 44(11) :116 –123, november 2006.

[LZH12] Na Li, Weihong Zhu, and Haihua Han. Digital interference cancellation in single channel, full duplex wireless communication. In *Wireless Communications, Networking and Mobile Computing (WiCOM), 2012 8th International Conference on*, pages 1–4, Sept 2012.

[MBAAP02] A. Munaretto, H. Badis, K. Al-Agha, and G. Pujolle. A link-state qos routing protocol for ad hoc networks. In *Mobile and Wireless Communications Network, 2002. 4th International Workshop on*, pages 222–226, 2002.

[MCL+01] P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot. Optimized link state routing protocol for ad hoc networks. In *IEEE International Multi Topic Conference, IEEE INMIC*, 2001.

[MD01] M.K. Marina and S.R. Das. On-demand multipath distance vector routing in ad hoc networks. In *Ninth International Conference on Network Protocols*, 2001.

[Mis04] Ajay R. Mishra. *Fundamentals of Cellular Network Planning and Optimisation : 2G/2.5G/3G ...Evolution to 4G*, volume 4, Issue 6. 19 nov. 2004.

[MM06] Mascolo C. Musolesi M. A community based mobility model for ad hoc network research. In *REALMAN '06 Proceedings of the 2nd international workshop on Multi-hop ad hoc networks : from theory to reality*, pages 31 – 38, 2006.

[MMN+] Jodi A. Miller, Paul D. Minear, Albert F. Niessner, Anthony M. DeLullo, Brian R. Geiger, Lyle N. Long, and Joseph F. Horn. Intelligent unmanned air vehicle flight systems. In *Journal of Aerospace Computing, Information, and Communication, Vol. 4, No. 5, 2007, pp. 816.*

[MRM11] F. Morbidi, C. Ray, and G.L. Mariottini. Cooperative active target tracking for heterogeneous robots with application to gait monitoring. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 3608–3613, Sept 2011.

[MYM01] M.P. Marios, Y. Yanli, and Kevin M.P. A cooperative search framework for distributed agents. In *Proceedings of the 2001 IEEE International Symposium on Intelligent Control, Mexico*, 2001.

[Pap] Paparazzi. http ://wiki.paparazziuav.org/wiki/Main_Page.

[PR99a] C.E. Perkins and E.M. Royer. Ad hoc on-demand distance vector routing. In *2nd IEEE Workshop on Mobile Computing Systems and applications*, 1999.

[PR99b] Charles E. Perkins and Elizabeth M. Royer. Ad hoc on demand distance vector (aodv) routing draft-ietf-manet-aodv-02.txt. IETF Draft, November 1999.

[PZ09] Aveek Purohit and Pei Zhang. Sensorfly : a controlled-mobile aerial sensor network. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, pages 327–328, 2009.

[QAKQ11] M.M. Qabajeh, A.H. Abdalla, O. Khalifa, and L.K. Qabajeh. A tree-based qos multicast routing protocol for manets. In *4th International Conference on Mechanisms (ICOM), IEEE*, 2011.

[QKWS+10] Markus Quaritsch, Karin Kruggl, Daniel Wischounig-Strucl, Subharata Bhattacharya, and Bernhard Rinner Mubarak Shah. Networked uavs as aerial sensor network for disaster management applications. *e & i Elektrotechnik und Informationstechnik*, 127, Issue 3 :56–63, 2010.

[RAV01] E. Rosen and R. Callon A. Viswanathan. Multiprotocol label switching architecture. IETF RFC 3031, Jan. 2001.

[Roy10] Radhika Ranjan Roy. *Handbook of Mobile Ad Hoc Networks for Mobility Models*. 2010.

[RPG13] M. Royer, A. Pirovano, and F. Garcia. Survey on context-aware publish/subscribe systems for vanet. In *5th Intl Workshop on Communication Technologies for Vehicles May 14-15, 2013 - Lille (France)*, 2013.

[RS07] Ed. R. Stewart. Stream control transmission protocol. IETF, RFC 4960, Septembre 2007.

[RZH+04] A. Ryan, M. Zennaro, A. Howell, R. Sengupta, and J.K. Hedrick. An overview of emerging results in cooperative uav control. In *43rd IEEE Conference on Decision and Control*, 2004.

[SFS+04a] P. Sholander, G. Frank, Sean Swank, J.P. Loyall, and G. Duzan. Multi-layer, mission-aware qos management techniques for ip applications in a joint battlespace infosphere. In *Military Communications Conference, 2004. MILCOM 2004. 2004 IEEE*, volume 3, pages 1254–1260 Vol. 3, Oct 2004.

[SFS+04b] P. Sholander, G. Frank, Sean Swank, J.P. Loyall, and G. Duzan. Multi-layer, mission-aware qos management techniques for ip applications in a joint battlespace infosphere. In *Military Communications Conference, 2004. MILCOM 2004. 2004 IEEE*, volume 3, pages 1254–1260 Vol. 3, Oct 2004.

[Shi11] R. Shirani. Reactive-greedy-reactive in unmanned aeronautical ad-hoc networks : A combinational routing mechanism. Master's thesis, Carleton University, available on : http ://www.csit.carleton.ca/˜msthilaire/Thesis/Rostam%20Shirani.pdf, 2011.

[SKKK03] J. Salim, H. Khosravi, A. Kleen, and A. Kuznetsov. Linux netlink as an ip services protocol. IETF, RFC 3549, Juillet 2003.

[Soc03] IEEE Computer Society. Ieee std. 802.15.4-2003, October . 2003.

[SSB99] P. Sinha, R. Sivakumar, and V. Bharghavan. Cedar : a core-extraction distributed ad hoc routing algorithm. In *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 1, pages 202–209 vol.1, Mar 1999.

[SSHK+11] R. Shirani, M. St-Hilaire, T. Kunz, Yifeng Zhou, Jun Li, and L. Lamont. The performance of greedy geographic forwarding

in unmanned aeronautical ad-hoc networks. In *Ninth Annual Communication Networks and Services Research Conference (CNSR)*, 2011.

[TB13] S. Temel and I. Bekmezci. On the performance of flying ad hoc networks (fanets) utilizing near space high altitude platforms (haps). In *Recent Advances in Space Technologies (RAST), 2013 6th International Conference on*, pages 461–465, June 2013.

[TBH04] S.Jadhav T.X. Brown, S. Doshi and J. Himmelstei. Test bed for a wireless network on small uavs. In *AIAA, 3rd Unmanned Unlimited Technical conference, Chicago, IL*, 2004.

[TBV13] Tan Viet Anh Truong, Petr Benda, and Jiri Vokrinek. The task agent resource function application in uav domain. In *AIAA GNC 2013, AIAA Guidance, Navigation and Control Conference, Boston : United States DOI : 10.2514/6.2013-4798*, (2013).

[TCP81] TCP. Transmission control protocol. IETF, RFC : 793, 1981.

[TH03] B. Tavli and W.B. Heinzelman. Mh-trace : multihop time reservation using adaptive control for energy efficiency. In *Military Communications Conference, 2003. MILCOM '03. 2003 IEEE*, volume 2, pages 1292–1297 Vol.2, Oct 2003.

[TH05] B. Tavli and W.B. Heinzelman. Mc-trace : multicasting through time reservation using adaptive control for energy efficiency. In *Military Communications Conference, 2005. MILCOM 2005. IEEE , pp.2672-2678 Vol. 4, 17-20 Oct*, 2005.

[TTV07] F. Theoleyre, R. Tout, and F. Valois. New metrics to evaluate mobility models properties. In *Wireless Pervasive Computing, 2007. ISWPC '07. 2nd International Symposium on*, pages –, Feb 2007.

[WAKP08] Y. Wang, A. Ahmed, B. Krishnamachari, and K. Psounis. Ieee 802.11p performance evaluation and protocol enhancement. In *IEEE International Conference on Vehicular Electronics and Safety, 2008. ICVES 2008. Sept*, 2008.

[WGWW10] Wei Wanga, Xiaohong Guana, Beizhan Wangb, and Yaping Wangc. A novel mobility model based on semi-random circular movement in mobile ad hoc networks. *Information Sciences*, Volume 180, Issue 3 :399–413, 2010.

[XHG+01] Kaixin Xu, Xiaoyan Hong, Mario Gerla, H. Ly, and D.L. Gu. Landmark routing in large wireless battlefield networks using

uavs. In *Military Communications Conference, 2001. MILCOM 2001. Communications for Network-Centric Operations : Creating the Information Force. IEEE*, 2001.

[XLH+13] T. Xie, R. Liu, R.T. Hai, Q.H. Hu, and Q. Lu. Uav platform based atmospheric environmental emergency monitoring system design. *Journal of Applied Sciences*, 13 :1289–1296, 2013.

[XSX+10] Ji Xiangyu, Wu Sentang, Liu Xiang, Du Yang, and Tang Jiqiang. Research and design on physical multi-uav system for verification of autonomous formation and cooperative guidance. In *Electrical and Control Engineering (ICECE), 2010 International Conference on*, pages 1570–1576, June 2010.

[XTP92] XTP. Xpress transfer protocol. XTP Forum, version 3.6, January 1992.

[XWNF13] J. Xie, Y. Wan, K. Namuduri, and J. Fu, S.and Kim. A comprehensive modeling framework for airborne mobility. In *The Americain Institute of Aeronautics and Astronautics (AIAA) Conference, Boston*, August 19-22, 2013.

[XYLC00] Hannan Xiao, Winston K. G. Seah Y, Anthony Lo, and Kee Chaing Chua. A flexible quality of service model for mobile ad-hoc networks. In *IEEE VTC2000-spring*, pages 445–449, 2000.

[YCBE10] E. Yanmaz, C. Costanzo, C. Bettstetter, and W. Elmenreich. A discrete stochastic process for coverage analysis of autonomous uav networks. In *GLOBECOM Workshops (GC Wkshps), 2010 IEEE*, pages 1777–1782, Dec 2010.

[YKB11] E. Yanmaz, R. Kuschnig, and C. Bettstetter. Channel measurements over 802.11a-based uav-to-ground links. In *GLOBECOM Workshops (GC Wkshps), 2011 IEEE*, pages 1280–1284, Dec 2011.

[YPD07] C. Yawut, B. Paillassa, and R. Dhaou. On metrics for mobility oriented self adaptive protocols. In *Wireless and Mobile Communications, 2007. ICWMC '07. Third International Conference on*, pages 11–11, March 2007.

[ZHR04] Qunwei Zheng, Xiaoyan Hong, and Sibabrata Ray. Recent advances in mobility modeling for mobile ad hoc network research. In *Proceedings of the 42Nd Annual Southeast Regional Conference*, ACM-SE 42, pages 70–75, New York, NY, USA, 2004. ACM.

Annexe A

Annexe : Outils de simulation

Le simulateur réseau OMNET++

OMNET++ (***O**bjective **M**odular **Ne**twork **T**estbed in C++*)[1] est une plateforme de simulation à événement libre développée en C++.

Il a été conçu pour simuler les réseaux de communication, les systèmes multiprocesseurs et d'autres systèmes distribués. Il peut également être étendu en modélisant d'autres systèmes. C'est un outil modulaire, générique, flexible et basé composants. Il offre une librairie de classes de simulation C++ et un support GUI (*Graphical User Interface*) pour la visualisation graphique du réseau ainsi que son animation. Il contient entre autres un compilateur *NED* (*Network description)*, une interface graphique (*Tkenv*), un invite de commande (*Cmdenv*), un outil graphique pour l'analyse des résultats de simulation (*Dataset*), etc (figure A.1).

L'architecture de simulation avec OMNET++ est composée de modules qui communiquent entre eux en échangeant des messages à travers leurs interfaces liées par des connections.

OMNET++ possède des extensions externes qui lui permettent de fournir un support pour la simulation des réseaux sans fil tels que le plateformes *INET*, *INETMANET* ou encore *MiXiM*. Pour notre étude nous avons utilisé la plateforme *INET*.

Le modèle de simulation de DAN

DAN ainsi que le modèle de mobilité PPRZM ont été implémentés et simulés avec OMENT++. Chaque nœud utilisant DAN contient dans son

1. http ://www.omnetpp.org/

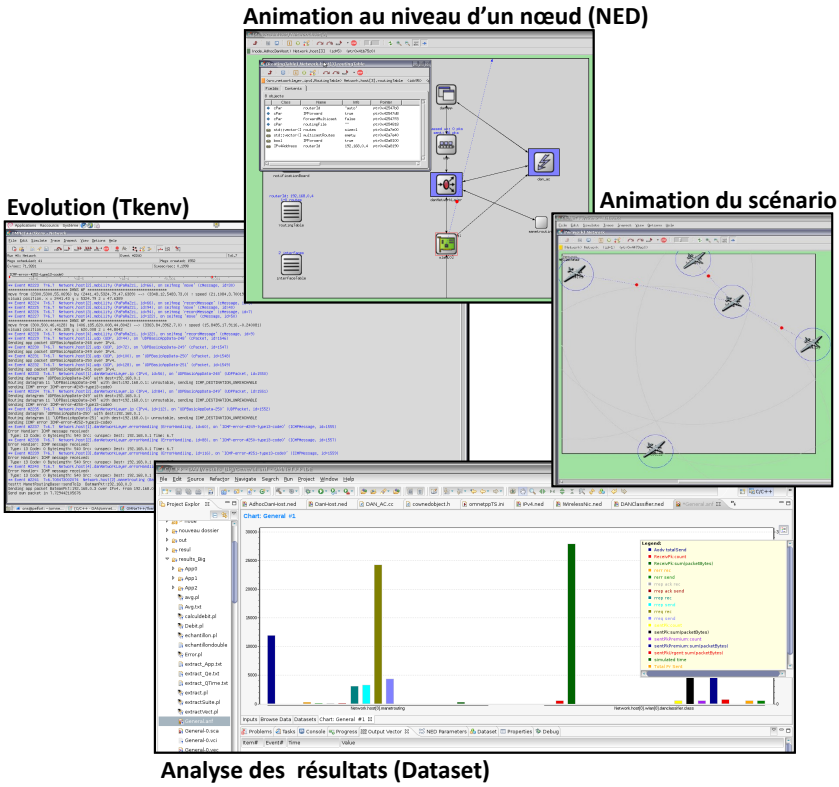


FIGURE A.1 – Captures écran du simulateur OMNET++

architecture le module *DAN-agent* comme présenté dans la figure A.2.

L'agent_DAN est lié aux niveaux réseau et applicatif qui peut émettre du trafic TCP et/ ou UDP. Chaque application peut communiquer avec DAN à travers DAN_API. Par la suite, l'agent_DAN communique directement avec les autres modules ainsi qu'avec les niveau réseau pour envoyer et recevoir des messages de signalisation encapsulés dans IP.

La figure A.3 présente l'architecture du module DAN_classificateur. Il contient un module classificateur (*class)* qui permet de marquer les trafic si le nœud est l'émetteur et d'affecter les paquets reçus à l'une de ses trois files d'attente : Urgent, Premium et BE.
Le type et la taille de la file étant des paramètres pour ce modèle, nous avons utilisé pour nos simulations le type FIFO (*First In First Out*) et la taille 100 (nombre maximum de paquets) par file.

Le module d'ordonnancement, nommé *Ps*, est responsable de retirer les paquets à émettre des files d'attentes. Ce module est un ordonnancement qui

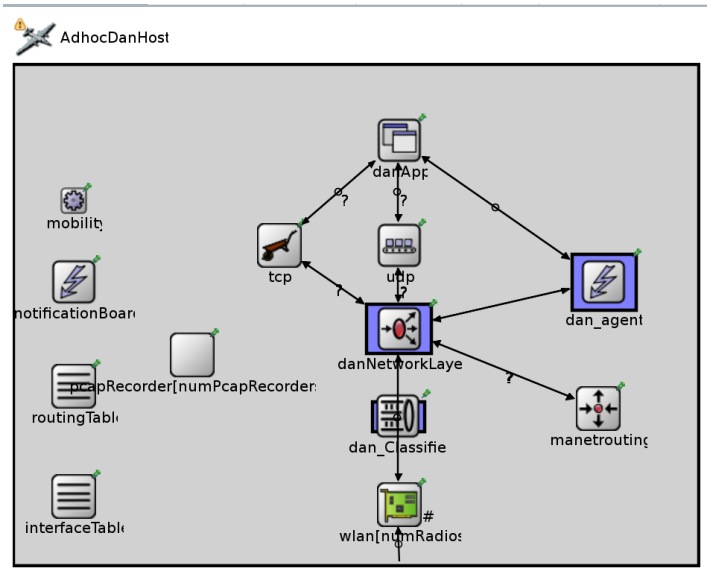


FIGURE A.2 – Architecture d'un nœud utilisant le système DAN

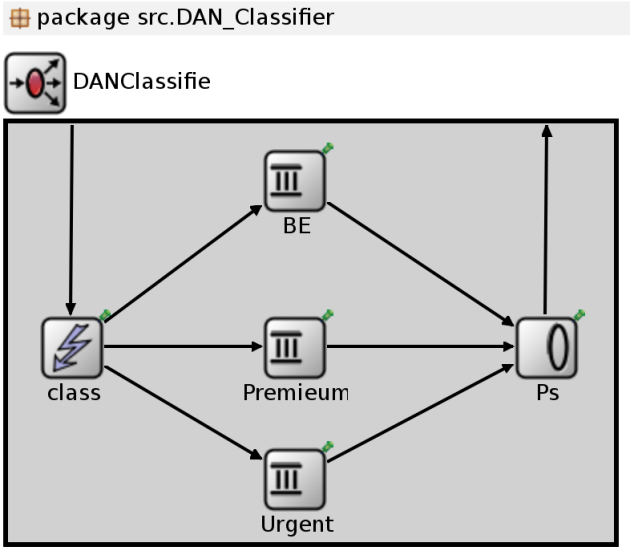


FIGURE A.3 – Architecture du module DAN_classificateur

permet de respecter les priorités des trois files d'attente (*Priority Queing*).

Pour nos simulations, nous avons utilisé le modèle de mobilité PPRZM créé pour imiter les mouvements réels de drones Paparzzi.

Le reste des modules utilisés pour l'architecture des nœuds sont les modules ordinaires proposé par la plateforme *INET*.