



UNIVERSITÉ DU QUÉBEC EN OUTAOUAIS

MÉMOIRE PRÉSENTÉ

COMME EXIGENCE PARTIELLE DU PROGRAMME
DE MAITRISE EN INFORMATIQUE

PAR
CHRISTIAN DECHAMPLAIN

ÉTUDE DE SOLUTIONS POUR L'OPTIMISATION DE
L'ÉQUITÉ ET LA QUALITÉ DE SERVICE DES FLUX
DANS LES RÉSEAUX VANET AVEC ADRESSAGE CONDITIONNEL

ÉTÉ 2015

Jury d'évaluation

Directeur de recherche: Dre Ilham Benyahia

Président du jury: Dr Luigi Logrippo

Membre du jury: Dr Ahmed Lakhssassi

Remerciements

Je remercie la professeure Ilham Benyahia pour sa disponibilité, ses conseils et toute l'aide qu'elle a fournie pour la rédaction de ce mémoire.

Je remercie également les membres du jury professeur Luigi Logrippo et professeur Hamed Lakhssassi pour leurs suggestions enrichissantes.

Je remercie également le professeur Mohamed Shawky pour sa visite et sa collaboration.

Enfin, Je remercie les membres de ma famille pour leur soutien.

Résumé

Les systèmes de transports intelligents ITS (*Intelligent Transport Systems*) [1] encouragent le développement de nouvelles applications dans différents champs des technologies tels que les réseaux de communications sans fil ad hoc et les réseaux de communications sans fil véhiculaires avec infrastructure. Les véhicules des ITS sont munis d'équipements de réseautage sans fil permettant la communication de véhicule à véhicule (V&V) et la communication de véhicule à infrastructure (V&I), la combinaison des deux types de réseaux forme des réseaux hybrides. Ces équipements supportent le protocole DSRC pour les communications dédiées à courte distance (*Dedicated Short Range Communications*). Le standard pour ces communications est celui de l'IEEE, 802.11p [2], qui utilise la bande radio large de 75 MHz située à 5.9Ghz. Ce protocole de la famille 802.11 est une variante des protocoles WIFI (*Wireless Physical*) utilisés dans les ordinateurs portables et différents appareils électroniques à communications sans fil de la vie de tous les jours.

Les protocoles existants qui sont utilisés initialement sur ces réseaux présentent des lacunes au niveau de la QoS (Qualité du Service) et de l'équité au niveau des flux de communication. En effet, les travaux scientifiques qui portent sur la QoS et l'équité dans un contexte VANET (*Vehicular Ad-hoc NETWORKS*) le font principalement au niveau des véhicules [3], alors que l'importance des flux est primordiale, car un véhicule peut servir de relais aux autres véhicules [4]. De plus, les travaux scientifiques qui portent sur la QoS et l'équité au niveau des flux le font dans un contexte de couches de réseaux supérieures, et font abstraction des défis des réseaux sous-jacents par exemple VANET. Ce qu'on propose ici c'est d'intégrer les concepts d'équité et de QoS dans l'adressage entre les véhicules selon un modèle intercouche afin d'optimiser le partage de la bande passante pour les communications, et en démontrer l'efficacité dans un scénario de congestion autoroutière avec réseau de communication VANET simulé.

Mots clés : réseaux véhiculaires, QoS, équité, optimisation, protocoles d'accès, protocoles de routage, files d'attente

Table des matières

Remerciements	5
Résumé.....	7
Table des matières.....	8
Table des figures	11
Abréviations	13
1 Introduction	16
2 Problématique	19
2.1 Objectif principal : gestion de la congestion de VANET avec l'équité et la QoS.....	21
2.2 Objectifs secondaires : intégration et optimisation de l'équité et la QoS avec un protocole de routage dynamique adéquat.....	22
3 État de l'art.....	24
3.1 Traitement au niveau des couches inférieures.....	24
3.1.1 Techniques CDMA et IDMA	24
3.1.2 Fonctions et protocoles de la couche MAC	25
3.1.3 Algorithmes de gestion de files d'attente	31
3.2 Couche réseau.....	35
3.2.1 Adressage conditionnel	35
3.2.2 Routage des réseaux ad hoc	38
3.3 Couches supérieures	39
3.3.1 Couche TCP	40
3.3.2 Couche application	40

3.4	Concepts d'architecture intercouches	41
3.4.1	Intelligence artificielle.....	42
3.4.2	Techniques intercouches	46
3.5	Synthèse	47
3.5.1	Algorithme EDCF	50
3.5.2	Algorithme hybride de gestion des files d'attente	52
3.5.3	Algorithme transmissions conditionnelles	54
4	Méthodologie.....	58
4.1	Objectif principal: L'équité et la QoS pour la gestion de la congestion des VANETs au niveau des flux de communication	58
4.2	Objectif secondaire : Algorithme de routage dynamique intégrant l'équité et la QoS.	59
4.3	Objectif secondaire : optimisation du routage dynamique avec équité et QoS.....	60
4.3.1	Architecture de gestion équitable des flux sur VANET.....	62
4.3.2	Algorithme de la FDCF	62
4.3.3	Algorithme hybride de gestion des files d'attente pigeonier stratifié	65
4.3.4	Algorithme des transmissions conditionnelles à trajet déterminé	67
4.4	Diagramme du modèles des couches.....	68
4.5	Présentation du système de simulation.....	69
4.6	Implémentation du scénario.....	71
4.7	Expérimentation.....	72
4.7.1	Mesures sur l'équité des flux.....	72
4.7.2	Adéquation de l'équité dans le modèle de QoS	72
4.7.3	Optimisation de l'équité et de la QoS.....	72

5	Réalisation.....	73
5.1	Configuration du système d'expérimentation, défis et solutions	73
5.2	Résultats d'expérimentation.....	75
5.2.1	Résultats sans l'algorithme des transmissions conditionnelles.....	77
5.2.2	Résultats avec l'algorithme des transmissions conditionnelles	87
6	Conclusion.....	99
7	Références	100
	Annexe A syntaxe du pseudo-code.....	105

Table des figures

Figure 1 schématisation des délais de l'EDCF	27
Figure 2 situation de nœud caché	29
Figure 3 triage des flux et files d'attente	32
Figure 4 marqueurs trois couleurs deux taux	34
Figure 5 automate de décision d'évitement de bouchon	37
Figure 6 diagramme de séquence d'évitement de bouchon	38
Figure 7 application de contournement	41
Figure 8 choix des technologies	49
Figure 9 algorithme EDCF	52
Figure 10 algorithme hybride de gestion des files d'attente	54
Figure 11 algorithme transmissions conditionnelles	56
Figure 12 algorithme FDCF	65
Figure 13 algorithme BTSFQ	67
Figure 14 algorithme des transmissions conditionnelles à trajet déterminées	68
Figure 15 diagramme de couches	69
Figure 16 extension du modèle de base	70
Figure 17 sortie de route	71
Figure 18 tableau des options de scenario	76
Figure 19 visualisation du scénario avec l'outil Vehicle Viewer	77
Figure 20 paquets par classe de service sans QoS sans transmissions conditionnelles	79
Figure 21 délais par classe de service sans QoS sans transmissions conditionnelles	79
Figure 22 paquets par classe de service avec QoS sans transmissions conditionnelles	80
Figure 23 délais par classe de service avec QoS sans transmissions conditionnelles	81
Figure 24 paquets par classe de service avec QoS et équité sans transmissions conditionnelles	82
Figure 25 délais par classe de service avec QoS et équité sans transmissions conditionnelles	82
Figure 26 paquets par types de flux sans QoS sans transmissions conditionnelles	83

Figure 27 délais par type de flux sans QoS sans transmissions conditionnelles	84
Figure 28 paquets par types de flux avec QoS sans transmissions conditionnelles.....	85
Figure 29 délais par type de flux avec QoS sans transmissions conditionnelles	85
Figure 30 paquets par types de flux avec QoS et équité sans transmissions conditionnelles	86
Figure 31 délais par type de flux avec QoS et équité sans transmissions conditionnelles	87
Figure 32 paquets par classe de service sans QoS avec transmissions conditionnelles	88
Figure 33 délais par classe de service sans QoS avec transmissions conditionnelles	89
Figure 34 paquets par classe de service avec QoS avec transmissions conditionnelles	90
Figure 35 délais par classe de service avec QoS avec transmissions conditionnelles.....	90
Figure 36 paquets par classe de service avec QoS et équité avec transmissions conditionnelles	91
Figure 37 délais par classe de service avec QoS et équité avec transmissions conditionnelles...	92
Figure 38 paquets par type de flux sans QoS avec transmissions conditionnelles	93
Figure 39 délais par type de flux sans QoS avec transmissions conditionnelles	93
Figure 40 paquets par type de flux avec QoS avec transmissions conditionnelles	94
Figure 41 délais par type de flux avec QoS avec transmissions conditionnelles.....	95
Figure 42 paquets par type de flux avec QoS et équité avec transmissions conditionnelles	96
Figure 43 délais par type de flux avec QoS et équité avec transmissions conditionnelles	96
Figure 44 influence sur les décisions des conducteurs.....	97

Abréviations

ACO	<i>Ant Colony Optimization</i>
AIFS	<i>Arbitration Inter Frame Space</i>
ATM	<i>Asynchronous Transfer Mode</i>
AODV	<i>Ad hoc On-Demand Distance Vector</i>
BE	<i>Best Effort</i>
BEB	<i>Binary Exponential Backoff</i>
BK	<i>BackGround</i>
BTSFQ	<i>Binsort Tiered Service Fair Queueing</i>
CDMA	<i>Code Division Multiple Access</i>
CSMA/CA	<i>Carrier Sense Multiple Access with Collision Avoidance</i>
CSMA/CD	<i>Carrier Sense Multiple Access with Collision Detection</i>
CBS	<i>Committed Burst Size</i>
CIR	<i>Committed Information Rate</i>
CTS	<i>Clear To Send</i>
CW	<i>Contention Window</i>
DARPA	<i>Defense Advanced Research Projects Agency</i>
DCF	<i>Distributed Coordination Function</i>
DIFS	<i>Distributed Inter-Frame Space</i>
DSRC	<i>Dedicated Short Range Communications</i>
DRR	<i>Deficit Round Robin</i>
EDCF	<i>Enhanced Distributed Coordination Function</i>
EEBL-F	<i>Emergency Electronic Brake Light with Forwarding</i>
EIFS	<i>Extended Inter Frame Space</i>
FDCF	<i>Fair Distributed Coordination Function</i>
FOCALE	<i>Foundation, Observation, Comparison, Action, and Learning Environment</i>
FWM	<i>Fair Wireless MAC</i>

GPS	<i>Generalized Processor Sharing</i>
IDM	<i>Intelligent Driver Model</i>
IEEE	<i>Institute of Electric and Electronic Engineers</i>
IETF	<i>Internet Engineering Task Force</i>
IDMA	<i>Interleave Division Multiple Access</i>
ITS	<i>Intelligent Transport Systems</i>
LAR	<i>Location Aware Routing</i>
MAC	<i>Media Access Control</i>
OFDM	<i>Orthogonal Frequency Division Multiplexing</i>
OLSR	<i>Optimized Link State Routing Protocol</i>
PBS	<i>Peak Burst Size</i>
PIR	<i>Peak Information Rate</i>
POS	<i>Packet Over SONET</i>
QoS	<i>Qualité de Service</i>
RFC	<i>Request For Comment</i>
RTP	<i>Real-time Transmission Protocol</i>
RTS	<i>Request To Send</i>
SCTP	<i>Stream Control Transmission Protocol</i>
SIFS	<i>Short Inter-Frame Space</i>
SLA	<i>Service Level Agreement</i>
SLAM	<i>Simultaneous Location And Mapping</i>
TCP	<i>Transmission Control Protocol</i>
TCP/IP	<i>Transmission Control Protocol / Internet Protocol</i>
trTCM	<i>two rate Three Color Marker</i>
TSFQ	<i>Tiered Service Fair Queueing</i>
UDP	<i>User Datagram Protocol</i>

VANET	<i>Vehicular Ad-hoc NETworks</i>
VI	<i>Vidéo</i>
VO	<i>VOice</i>
V&I	Communications Véhicule à Infrastructure
V&V	Communications Véhicule à Véhicule
WAVE	<i>Wireless Access in Vehicular Environments</i>
WFQ	<i>Weighed Fair Queuing</i>
WiFi	<i>Wireless Physical</i>
WRED	<i>Weighed Random Early Discard</i>
XML	<i>eXtensible Markup Language</i>

1 Introduction

Les systèmes de transports intelligents ITS (*Intelligent Transport Systems*) [1] permettent de développer de nouvelles applications dans différents champs des technologies tels que les réseaux de communications sans fil véhiculaires ad hoc et les réseaux de communications sans fil véhiculaires avec infrastructure. Dans ces réseaux, les véhicules sont munis d'équipements de réseautage sans fil permettant la communication de véhicule à véhicule (V&V) et la communication de véhicule à infrastructure (V&I). Les communications de véhicules à véhicules se font entre des unités montées sur les véhicules alors que les communications de véhicules à infrastructure se font entre des unités montées sur les véhicules et des unités placées sur les bords des autoroutes [5]. La combinaison des deux types de réseaux forme des réseaux de communications hybrides, mais la présence d'infrastructure n'est pas nécessaire pour certaines applications comme la gestion du trafic. Ces équipements supportent le protocole DSRC pour les communications dédiées à courte distance (*Dedicated Short Range Communications*). Le standard pour ces communications est celui de l'IEEE, 802.11p [2], qui utilise la bande radio large de 75 MHz située à 5.9Ghz. Ce protocole de la famille 802.11 est une variante des protocoles WIFI (*Wireless Physical*) utilisés dans les ordinateurs portables et différents appareils électroniques à communications sans fil de la vie de tous les jours. La fonction de coordination distribuée DCF (*Distributed Coordination Function*) du protocole permet le partage de la bande passante sans qu'une station de base qui fait partie d'une infrastructure filaire ait besoin de coordonner les communications, ce qui permet la création de réseaux ad hoc et hybrides d'une nature dynamique sans nécessiter d'infrastructure de bord de route couvrant la totalité du réseau autoroutier.

L'aspect dynamique des réseaux de communications véhiculaires représente un défi qu'on ne retrouve pas dans les réseaux sans fil à topologie statique. Par exemple, le routage des paquets ne peut se fier à une infrastructure fixe pour trouver le meilleur chemin pour ces paquets et le

partage de la bande passante ne peut pas se faire simplement en augmentant les délais comme peuvent le faire les commutateurs des réseaux filaires. Ces problèmes de routage et de partage de bande passante sont en partie réglés par les technologies des couches des niveaux inférieurs décrites précédemment, et en partie réglés par de nouveaux protocoles de routage de haut niveau comme on peut voir dans [6] et dans [7]. Ces solutions vont du découpage de la bande passante en fréquences réservées pour la sécurité, le contrôle et les communications diverses, en passant par les algorithmes qui contournent les trous de trafic routier pour acheminer les paquets, en allant jusqu'aux systèmes qui font recours à l'intelligence artificielle pour optimiser le routage des paquets. À mesure que les nouvelles applications feront un usage grandissant de la bande passante, ces nouvelles technologies feront face à des problèmes de congestion du réseau de communication. Ces situations de congestion sont bien décrites dans [8]. Les échanges prioritaires de données, comme définis dans le modèle des services différenciés [9] qui classifie les communications dans différents niveaux de services, bénéficient d'un ensemble de fonctions prévues dans l'architecture du réseau garantissant la qualité de service [1], et même de canaux dédiés [2] qui permettent d'éviter les situations de congestion pour les messages des services d'urgence. Les échanges moins prioritaires du type meilleur effort (*best effort*) n'ont pas le même avantage donné par la priorité et voient leurs paquets se faire rejeter quand les réseaux sont congestionnés [10].

Pour ces échanges, de nature dynamique, qui ne bénéficient pas de priorités attribuées par intervention humaine sur la configuration du réseau, le concept qui présente un fort potentiel d'adéquation est l'équité comme décrite dans [11]. Le concept d'équité se définit comme suit; dans les situations de congestion, les flux qui utilisent moins que leur juste part de bande passante devraient avoir droit aux meilleurs délais, alors que les flux qui utilisent ou tentent d'utiliser plus que leur juste part de bande passante devraient se contenter de délais moins avantageux, et ces délais se doivent d'être bornés. Le fait même que les paquets de classe meilleur effort n'ont pas une priorité attribuée par configuration ne diminue en rien le niveau d'importance relative de ces paquets, et les flux de communication qui sont formés par ces

paquets pourraient bénéficier de l'équité pour les décisions de routage dans les cas de congestion du réseau.

Le document qui suit mettra en contexte la problématique des communications dans les réseaux VANET (*Vehicular Ad-hoc NETWORKS*) et comment ils peuvent bénéficier d'une meilleure équité pour optimiser la qualité du service au chapitre 2. L'état de l'art sera présenté dans le chapitre 3 pour appuyer le contexte et situer les travaux existants et situer les travaux existants par rapport à notre problématique. Par la suite, la méthodologie sera décrite ainsi que l'environnement d'expérimentation pour valider notre solution sera présenté au chapitre 4. Finalement on fera rapport de la réalisation au chapitre 5 et la conclusion viendra au chapitre 6.

2 Problématique

Les problèmes de congestions font partie d'une réalité des réseaux de communications où l'on n'effectue pas systématiquement de réservation pour la bande passante, que ces réseaux soient filaires ou pas. Ce principe s'applique aussi de façon générale, car même un réseau où l'on peut faire des réservations peut être congestionné [12]. Vu la mobilité des usagers sur un réseau véhiculaire, il ne serait pas pratique d'effectuer des réservations dynamiques des ressources de transmission. Notons tout de même que certaines réservations statiques sont faites, l'exemple parfait se trouve dans la spécification de DSRC avec les canaux réservés pour le contrôle et la sécurité [2]. En fait, deux véhicules en mouvement se trouveront parfois dans la possibilité de communiquer entre eux seulement pendant une courte durée, surtout si ce sont des véhicules se déplaçant dans des directions différentes, voire contraires, alors il pourrait être futile et même inefficace de tenter de réserver la bande passante dans ce type de cas. De plus, une des applications les plus intéressantes des réseaux routiers intelligents est la communication qui permettra d'éviter la congestion au niveau du trafic des véhicules sur le réseau routier lui-même, mais on peut supposer que de telles communications pourront entraîner ou contribuer à la congestion virtuelle des paquets sur le réseau de communication. L'article [8] décrit quelques-unes de ces situations de congestion des paquets, où la densité de la topologie véhiculaire augmente proportionnellement la probabilité de congestion en général. En effet, on peut considérer un scénario où tous les véhicules dans un bouchon de circulation essaient de communiquer avec les véhicules ayant encore la possibilité de prendre une route alternative, pour les avertir du bouchon. Dans ce type de situation, il serait facile de calculer un nombre v de véhicules dans une zone bouchon, essayant tous d'envoyer des messages de taille m , qui au total dépassent la bande passante disponible b dans un temps t , créant ainsi une situation simple de congestion de paquets. Ceci se traduit par l'équation suivante :

$$\sum_{i=0}^v m_i > \frac{b}{t}$$

En pratique, de nombreuses technologies implémentées aux différents niveaux des protocoles de communication vont permettre d'éviter de façon temporaire les situations de congestion, et même permettre la congestion seulement de façon temporaire, mais il ne sera jamais possible d'éliminer complètement les situations de congestion, comme décrites dans les articles qui traitent des variantes de gestion de la congestion au niveau de la couche TCP (Vegas, Reno, Tahoe, etc.) [13, 14]. À moins qu'on ne tente aucunement d'utiliser efficacement la bande passante disponible, ce qui ne peut être envisageable que de façon temporaire à mesure que la technologie est déployée. La demande croissante de bande passante venant des nouvelles applications développées confirme cet énoncé. Parmi ces technologies d'évitement temporaire, il y a les protocoles de niveau physique incluant les différentes technologies d'encodage comme CDMA et IDMA [15], les protocoles d'accès au média comme DSRC et WiFi [2, 3] avec les fonctions DCF et EDCF incluant les files d'attente et leurs algorithmes de gestion comme WFQ et DRR [4, 16] pour l'équité, les protocoles de routages incluant les modèles de qualité de service (QoS) comme DiffServ et IntServ [9]. La QoS est un concept selon lequel les flux classifiés de communication prioritaires ont un accès à la bande passante privilégié et conséquemment de meilleurs délais que les flux de communication classifiés de moindre priorité. Les solutions proposées dans la littérature sont toutes adéquates, tel qu'envisagées dans leurs contextes spécifiques. Pour les réseaux VANET, il faut aller plus loin par contre. Par exemple, les solutions de la couche 1 et de la couche 2 ne considèrent que les véhicules pour l'équité et la QoS. Les solutions de la couche 3 ne considèrent que les situations de routage. Les solutions des couches 4 et des autres couches supérieures ne considèrent que les flux et ignorent les outils et le contexte des couches inférieures.

2.1 Objectif principal : gestion de la congestion de VANET avec l'équité et la QoS

Tous ces efforts pour amoindrir les effets de la congestion sont justifiés, les chercheurs académiques et industriels travaillent à faire converger ces efforts afin de choisir les meilleures solutions sans jamais prétendre qu'il est possible de complètement éliminer les cas de congestion, il faut vivre avec et implanter les solutions acceptées par la communauté. Le défi qu'apportent les réseaux VANET avec leur nature dynamique augmentée par la mobilité des stations ajoute à la complexité de la gestion de la congestion. Les solutions classiques de configuration statique de la qualité de service qui sont utilisées dans les réseaux filaires, ou même sans fil avec infrastructure sont difficilement applicables avec les réseaux véhiculaires. Comme décrit dans les articles qui portent sur les transmissions conditionnelles, il est évident que l'adressage traditionnel n'est pas souvent adéquat dans les réseaux véhiculaires même s'il peut tout de même l'être dans quelques cas particuliers. De même, on peut faire l'hypothèse que la configuration traditionnelle qui accorde la meilleure qualité de service à certains types de communications ne sera pas toujours adéquate pour les réseaux véhiculaires ad hoc et hybrides. Le principe du « best-effort » s'applique bien dans les réseaux filaires ou sans-fils avec infrastructure qui peuvent plus facilement tolérer les pertes de paquets, ceux-ci bénéficient de la stabilité de l'infrastructure pour augmenter la probabilité de retransmission ou amoindrir l'impact d'une perte. Avec la nature dynamique des réseaux VANET, un véhicule pourrait très bien passer d'une zone où le médium est congestionné, à une zone où il n'aura plus d'interlocuteur pour relayer les messages, en l'espace de quelques secondes [6]. De même, un véhicule pourrait constamment émettre des messages, empêchant les autres véhicules d'avoir une probabilité raisonnable d'émettre quoi que ce soit, tant qu'ils sont dans cette zone d'émission considérée comme congestionnée. Même si ce véhicule n'a rien à émettre en vérité, il est très possible qu'il serve de relais aux autres véhicules qui transmettent abondamment. Ceci mène à notre objectif principal qui consiste à déterminer une technique pour identifier les messages dans un réseau VANET qui devraient avoir priorité sur les autres,

sans leur affecter une priorité statique, mais en favorisant un modèle de service équitable. L'équité est le concept en télécommunication selon lequel les messages des flux qui utilisent moins de bande passante que leur juste part se voient accordés des délais bornés et moindres que ceux des flux qui utilisent plus que leur juste part de bande passante.

Pour mettre en contexte cet objectif principal, nous envisageons le scénario décrit préalablement, celui où des véhicules sont soit dans une situation de bouchon de circulation sans pouvoir l'éviter ou soit dans une situation de circulation libre et en ayant encore une chance d'éviter le bouchon de circulation en prenant une sortie de route pour le contourner. Les véhicules dans la situation de bouchon inévitable pourront communiquer entre eux et avec l'infrastructure afin de détecter la situation de bouchon avec des paramètres de localisation et de vitesse par rapport à la structure du réseau routier. Les véhicules dans la situation de circulation libre pourront communiquer avec les véhicules du bouchon, soit directement ou soit relayés par d'autres véhicules et par l'infrastructure afin de connaître la situation de bouchon et ses paramètres de localisation et de vitesse par rapport à la structure du réseau routier.

2.2 Objectifs secondaires : intégration et optimisation de l'équité et la QoS avec un protocole de routage dynamique adéquat

Les communications VANET doivent se faire avec une certaine qualité de service c'est-à-dire une priorité entre les messages, et une équité des flux de communication sans priorité préétablie. L'équité et la priorité sont pratiquement des concepts diamétralement opposés, un réseau qui supporterait seulement la priorité stricte serait en fait assez inéquitable, alors qu'un réseau qui ne supporterait que l'équité pourrait ne pas convenir aux besoins en qualité de service des utilisateurs. Il faut donc pouvoir accorder les deux concepts complémentaires afin d'obtenir un réseau avec qualité de service et équité de façon équilibrée. Ceci nous amène un premier objectif secondaire : fournir à la fois la qualité de service et l'équité dans les réseaux VANET avec un protocole de routage dynamique adéquat. Notre second objectif secondaire consiste à optimiser la qualité de service et l'équité.

Même si nos protocoles implémentent des solutions pour fournir autant la qualité de service et l'équité, il n'est pas possible de prévoir toutes les situations pour les véhicules et les flux de communication. Ceci nous amène à notre dernier objectif secondaire : gérer ces situations imprévues d'utilisations de la bande passante.

La littérature scientifique expose amplement les problèmes reliés à la gestion de la qualité de service, de l'équité et de la sécurité pour les réseaux VANET. Par exemple, un survol de l'état de l'art est fait dans [17]. Cet article expose quelques solutions apportées par la science pour la qualité de service par exemple. Ces solutions et d'autres encore seront discutées dans le prochain chapitre de l'état de l'art. Notre analyse des solutions existantes pour l'équité et la qualité de service souligne l'emphase qui est mise sur les véhicules, alors qu'à la base l'importance se situe au niveau des flux de communication. Pour un exemple concret, prenons l'article [18] qui propose de gérer l'équité en utilisant la vitesse des véhicules impliqués comme paramètre pour déterminer qu'un message aura priorité sur un autre, étant donné que la vitesse relative entre deux nœuds affectera le temps disponible pour recevoir un message. Cette solution s'avère intéressante pour gérer l'équité entre les véhicules, mais en ne regardant les messages qu'au niveau des véhicules, sans se soucier des flux de communication comme tels, l'équité n'est pas optimisée.

3 État de l'art

Nous présentons dans ce chapitre, un état de l'art sur les travaux qui traitent sur les problématiques reliées à la congestion dans les réseaux de communication sans fil. Certaines contributions issues de ces travaux s'appliquent dans des contextes qui ne sont pas limités à la problématique de la congestion sur les réseaux VANET, et ainsi représentent des solutions générales du domaine de l'informatique et des communications de données. D'autres articles par contre traitent de scénarios qui sont spécifiques aux réseaux VANET comme le nôtre.

À partir de l'étude des travaux en lien avec notre projet nous avons en identifié trois catégories. La première catégorie concerne les couches inférieures du modèle de communication TCP/IP. La catégorie suivante concerne les couches supérieures du même modèle. La dernière catégorie concerne les concepts d'architecture comme l'intelligence artificielle et les techniques intercouches.

3.1 Traitement au niveau des couches inférieures

Rappelons que les couches inférieures sont composées de la couche physique (1), la couche d'accès au média (2) et la couche transport (3) du modèle TCP/IP. L'analyse de l'état de l'art pour les couches 1, 2 et trois est divisée en trois sections, soit la première pour les techniques CDMA/IDMA, la seconde pour les fonctions et protocoles de la couche MAC comme telle et la dernière pour les algorithmes de gestion de files d'attente.

3.1.1 Techniques CDMA et IDMA

Dans [15], on présente une nouvelle façon de faire l'encodage au niveau de la couche physique pour les réseaux sans fil ad hoc. La technique classique CDMA permet le partage de la bande radio en définissant que chaque utilisateur encode ses bits transmis avec un code déterminé au hasard. C'est donc une technique de partage de la bande passante par division de code. Ce mécanisme de code déterminé au hasard permet aussi de détecter à l'aide d'une fonction de

corrélation les situations dans lesquelles plusieurs utilisateurs sont en situation de collision sur le média. IDMA améliore ce mécanisme en s'assurant qu'il y ait une certaine distance entre les codes dépendamment du nombre d'utilisateurs présents, donc pas totalement au hasard. Il est possible qu'une couche physique de communication IDMA donne une meilleure équité que CDMA, mais les travaux présentés dans [2] montrent que la modulation choisie pour DSRC (802.11p) est l'OFDM (*Orthogonal Frequency Division Multiplexing*), qui a au moins une dimension de plus que CDMA et IDMA. En effet, d'autres moyens que la division de code sont utilisés pour partager l'air en tant que média de communication sans fil, il s'agit de la division de temps et la division de fréquence. La division de fréquence est clairement utilisée dans le protocole 802.11p, avec son canal réservé aux services d'urgence, ses deux canaux de contrôle et les quatre canaux de communication. Les travaux qui traitent de ce sujet sont intéressants, mais ce bas niveau n'est pas dans notre sphère d'influence pour l'amélioration de l'équité et la qualité de service. De tels travaux se situent plutôt au niveau du traitement de signaux que de l'informatique.

3.1.2 Fonctions et protocoles de la couche MAC

Dans [1], la fonction DCF (*Distributed Coordination Function*) est le mécanisme fondamental de la famille des protocoles d'accès et de contrôle de média 802.11. Cette fonction en est une d'accès aléatoire au média pour tous les nœuds dans un réseau ad hoc. Elle est basée sur la technologie CSMA/CA (*Carrier Sense Multiple Access with Collision Avoidance*, accès multiple à détection de porteuse avec évitement de collision), qui est une évolution de la technologie CSMA/CD (*Carrier Sense Multiple Access with Collision Detection*, accès multiple à détection de porteuse avec détection de collision) des communications filaires. En effet la détection de collision n'étant pas praticable pour le nœud émetteur radio, parce qu'il ne recevrait simplement que ce qu'il transmet en écoutant le signal. Une méthode proactive qui permet d'éviter des collisions est donc nécessaire pour les communications sans fil. Le mécanisme utilise l'échange de trames de contrôle CTS (*Clear To Send*, prêt pour l'envoi) et RTS (*Request To Send*, demande d'envoi), avec accusé réception pour minimiser le nombre de collisions. Dans le

cas ou les collisions arriveraient quand même, principalement dû à des nœuds cachés, c'est-à-dire des nœuds qui ne sont pas positionnés pour recevoir le signal d'un l'émetteur et qui se mettent à transmettre quand même, un mécanisme existe. Ce mécanisme de *backoff* (retrait temporaire) binaire exponentiel est utilisé avant que les stations puissent retransmettre. Ce temps d'attente, avec une composante de hasard ajoutée, permet d'éviter que les trames entrent en collision à nouveau.

Au niveau des protocoles d'accès et contrôle du média sans fil, pour l'amélioration de la contention du média utilisable par une station émettrice dans un rayon d'action immédiat, on retrouve la fonction de coordination distribuée améliorée EDCF (*Enhanced Distributed Coordination Function*) qui permet de partager le média en adaptant les délais d'attente avant chaque transmission selon la qualité de service et la priorité des paquets. À ce niveau la congestion devient pratiquement synonyme de collision, en effet la congestion aggravera les possibilités de collisions parce que la bande passante devient de plus en plus demandée. Les émetteurs nombreux, et possiblement nouveaux sur le réseau, dépasseront les capacités de partage de la bande passante et auront besoin de transmettre en même temps sur les mêmes canaux avec des codes identiques. Les files d'attente de transmission s'emplieront possiblement jusqu'au point où des paquets devront être oubliés, parce qu'il n'y aura plus de place pour eux sur les files d'attente. Ces files d'attente doivent être gérées de façon à prioriser les paquets ayant des requis de meilleure qualité de service par rapport aux autres paquets de moindre priorité, ce qui toutefois ne règle pas complètement le problème de congestion, mais au moins garde les délais raisonnables pour les communications prioritaires. Dans les travaux sur la performance de l'EDCF [3], on décrit un contexte qui aura un impact sur l'équité, parce que celui-ci touche la gestion de la bande passante et les délais d'accès. L'EDCF y est décrite comme suit. Il s'agit d'une extension de la DCF (*Distributed Coordination Function*) avec un meilleur partage du média radio sans fil par plusieurs utilisateurs en introduisant des délais d'attente de transmission selon la priorité assignée quand le lien radio est occupé. Ces délais sont déterminés par des fenêtres de contention composée d'éléments de temps standards de

la DCF avec des suppléments temporels qui sont inversement proportionnels aux niveaux de priorités. En fait l'EDCF ajoute une dimension QoS à la fonction de délai d'attente, dépendamment des classes de service. Il s'agit d'une approche prometteuse basée sur l'extension de la DCF qui peut avoir beaucoup d'impact sur l'équité de DSRC au niveau des véhicules.

La Figure 1 illustre les différents temps d'attente des fonctions DCF et EDCF. Le canal doit être libre pour au moins une période DIFS (*Distributed Inter-Frame Space*, espace intertrame distribué) avant de transmettre la trame. Une fois la trame transmise, le récepteur doit attendre au moins une période SIFS (*Short Inter-Frame Space*, espace intertrame court) avant de transmettre la trame de contrôle accusé réception. Le transmetteur, une fois l'attente DIFS terminée, doit en plus attendre le temps de retrait BEB (*Binary Exponential Backoff*, retrait binaire exponentiel), dans le cas de la DCF. Dans le cas de l'EDCF, un délai d'attente supplémentaire associé à la classification de qualité de service est ajouté, ce qui permet aux stations avec des messages prioritaires d'utiliser le canal pendant ce temps d'attente supplémentaire. L'effet net est de diminuer la performance globale, vu les temps d'attente supplémentaires, mais la qualité de service est améliorée. Par conséquent, la performance peut être améliorée dans les cas des transmissions prioritaires, mais diminuées dans le cas des transmissions « meilleur effort ». C'est un compromis acceptable étant donné l'importance de la qualité de service.

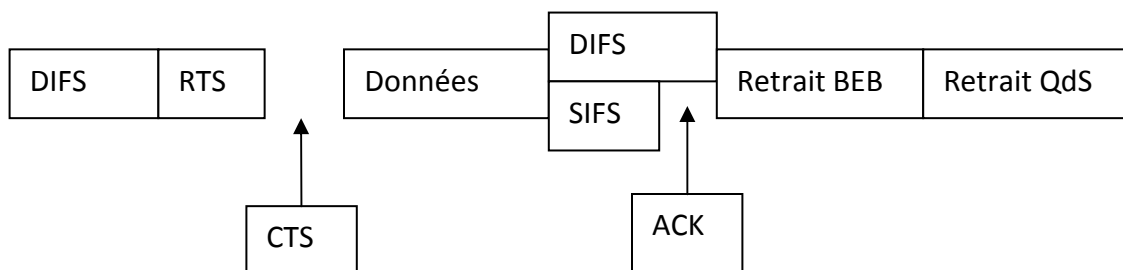


Figure 1 schématisation des délais de l'EDCF

Les travaux sur l'équité des réseaux ad hoc WiFi [19] présentent des scénarios qui sont des cas typiques d'iniquité très spécifiques aux réseaux sans fil de la famille 802.11 avec une topologie relativement statique, c'est-à-dire avec des nœuds qui sont cachés ou exposés. Les nœuds cachés peuvent créer des situations d'iniquité et les nœuds exposés peuvent être les victimes des situations d'iniquité. L'approche présentée propose une solution, appelée FWM (*Fair Wireless MAC*), qui en plus de la DCF standard, utiliserait deux canaux supplémentaires au canal de réception et de transmission. Un de ces canaux serait utilisé pour transmettre un signal indiquant aux autres nœuds quand un nœud est à l'écoute, afin d'éviter les collisions avec les nœuds cachés qui ne peuvent recevoir le signal du transmetteur, mais qui pourraient brouiller le signal du receveur. L'autre canal serait utilisé pour transmettre un signal de resynchronisation quand un nœud doit utiliser la fonction EIFS (*Extended Inter Frame Space*) avant de retransmettre, afin que toutes les stations utilisent le même point de départ pour la fonction DIFS et en plus retransmettre le signal de resynchronisation. L'efficacité de la solution est démontrée par simulation pour deux scénarios particuliers qui généralisent les scénarios possibles. L'équité qui est analysée dans l'article se situe au niveau des nœuds, mais aucune solution n'est présentée pour les flux de communication individuels. En effet une telle solution pourrait s'avérer temporairement inéquitable au niveau des flux dans un réseau à nature plus dynamique que dans le cas particulier observé dans ces travaux. La Figure 2 illustre le problème des nœuds cachés. Le nœud A peut transmettre au nœud B, mais le nœud C est hors de portée de cette conversation et peut donc tenter de communiquer avec B, créant ainsi une collision. Le nœud B n'accusera pas réception pour la trame provenant du nœud A. Le nœud A enclenchera le mécanisme de retrait en augmentant la grandeur de la fenêtre de contention, même chose pour le nœud C. Les cercles pointillés représentent les portées respectives des signaux des nœuds.

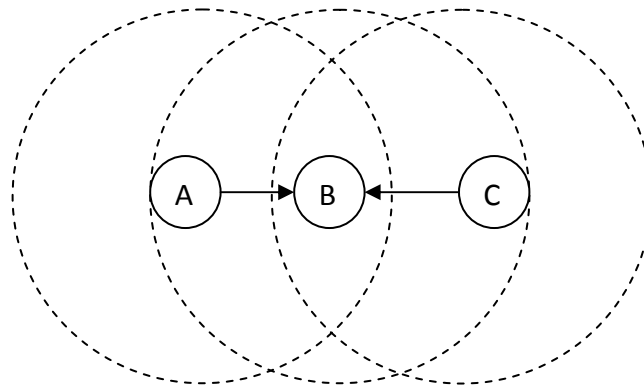


Figure 2 situation de nœud caché

Les solutions présentées dans [20], ont pour but d'aider à contrôler la congestion sur le canal de sécurité des protocoles de communication sans fil 802.11p dans un contexte d'utilisation de la technologie DSRC dont les amendements aux protocoles de la famille 802.11 sont communément appelés WAVE (*Wireless Access in Vehicular Environments*). Les performances de ces solutions sont validées par simulation avec le système WARP2. Le protocole 802.11p prévoit 7 canaux pour la communication dont un réservé pour les applications de sécurité routière. Même si le canal est réservé, donc toute la bande passante est disponible aux applications de sécurité routière, il est possible que des cas de congestion surviennent quand même sur ce canal. La QoS sur ce canal doit donc être garantie pour les applications de sécurité prioritaires. Dans ces travaux, l'application EEBL-F (*Emergency Electronic Brake Light with Forwarding*) est utilisée comme exemple. Cette application utilise le canal de sécurité de 802.11p pour relayer l'information sur les freinages brusques des véhicules en avant. Cette application est justifiée en cas d'accident pour que les conducteurs des véhicules s'approchant de la collision soient avertis plus vite que par les avertisseurs de freinage des véhicules devant eux. Si le canal de communication était congestionné, cette application ne pourrait relayer l'information à temps et dans ce cas il est même possible que les avertisseurs de freinage visuels classiques soient plus rapides à avertir les conducteurs que l'application EEBL-F. Dans ce cas les auteurs ont choisi de travailler avec la couche 2, la couche de contrôle et d'accès au

média, notamment la couche qui permet de gérer la qualité de service en offrant un nombre de files d'attente de transmission avec différentes priorités. Par ces moyens on peut aussi contrôler la puissance de transmission et changer les paramètres de gestion de la DCF, comme les différents temps d'attente définis précédemment. C'est sur ces paramètres, la priorité, les délais et la puissance de transmission, que les solutions de cet article sont basées. Donc ces travaux proposent comme première solution que les véhicules doivent constamment évaluer l'utilisation du canal de sécurité et réserver une partie de la bande passante du canal. C'est-à-dire agrandir la fenêtre de contention (la doubler) quand le taux d'utilisation dépasse un certain pourcentage (70 %), et diminuer (de moitié) la fenêtre de contention quand le taux d'utilisation est en deçà d'un certain pourcentage (30 %). Les travaux ont aussi démontré l'efficacité de cette solution par simulation. En conclusion, cette approche de qualité de service dynamique répond bien aux contraintes de l'application EEBL-F dans les cas de congestion. De plus, comme deuxième solution, si jamais le taux d'utilisation devait dépasser un seuil critique (95 %), dans ce cas on propose de bloquer toutes les files d'attente de transmission, pour s'assurer qu'il reste de la bande passante pour l'application EEBL-F. L'essai de cette méthode par simulation n'a pas donné des résultats aussi bons que pour la première solution, et l'impact sur le reste du trafic est assez négatif. Par le même mécanisme de simulation, il a aussi été vérifié à savoir si l'augmentation de la puissance d'émission pourrait avoir un effet en particulier pour l'application de sécurité EEBL-F, comme candidat pour une troisième solution. Il a été conclu par contre que l'augmentation de la puissance pour les applications de sécurité n'a pas d'effet bénéfique sur la qualité de service. Dans les scénarios qui nous intéressent, le canal de sécurité ne sera pas utilisé, car on veut faire la promotion de communications non urgentes, mais importantes et caractérisées par leur profil d'utilisation de la bande passante.

3.1.3 Algorithmes de gestion de files d'attente

Dans l'article [4] un classique qui définit l'algorithme WFQ (*Weighed Fair Queuing*), on définit l'algorithme référentiel d'ordonnancement des files d'attente pour fournir l'équité selon le poids donné aux files d'attente. Cet algorithme modélise l'algorithme idéal GPS (*Generalized Processor Sharing*), appliqué particulièrement à l'ordonnancement des files d'attente. La contribution de cet article est énorme parce que l'algorithme WFQ qu'il y est défini est la référence reconnue quand il s'agit de définir des algorithmes d'ordonnancement de files d'attente. En effet même si cet algorithme est idéal, il l'est d'un point de vue purement théorique parce qu'il est trop coûteux à implémenter à large échelle. Il demande à ce que tous les flux de communication aient leur propre file d'attente, ce qui est déjà très coûteux, mais en plus l'algorithme fait un tri des files d'attente à chaque fois qu'un élément doit en être sorti. Pour cette raison, de nombreux algorithmes comme celui considéré comme un standard dans l'industrie des télécommunications, le décrit dans l'article [16]. Cet algorithme présente des failles par contre, et plusieurs chercheurs travaillent à trouver un autre algorithme qui sera le standard de l'industrie. Par exemple l'algorithme DRR (*Deficit Round Robin*) présente des cas d'iniquité pour les flux de communication qui utilisent très peu de bande passante. Déjà dans l'article présentant l'algorithme DRR on avait prévu les problèmes potentiels et on a proposé DRR+, mais sans toutefois tenter de démontrer sa validité. D'autres articles documentent des cas d'iniquité par exemple dans l'article [21], ou on peut constater que les longs segments sont favorisés.

Beaucoup d'autres algorithmes ont été proposés par la communauté scientifique, et ceux qui nous intéressent sont ceux de la famille des algorithmes hybrides. Ces algorithmes font une simplification du problème de tri de l'algorithme WFQ en triant les flux dans un pigeonier par rapport à leur utilisation de la bande passante et utilisent la grandeur des paquets en tête de file, pour ordonnancer les files et garantir des délais bornés pour les flux [22]. Cette méthode est schématisée en Figure 3 triage des flux et files d'attente.

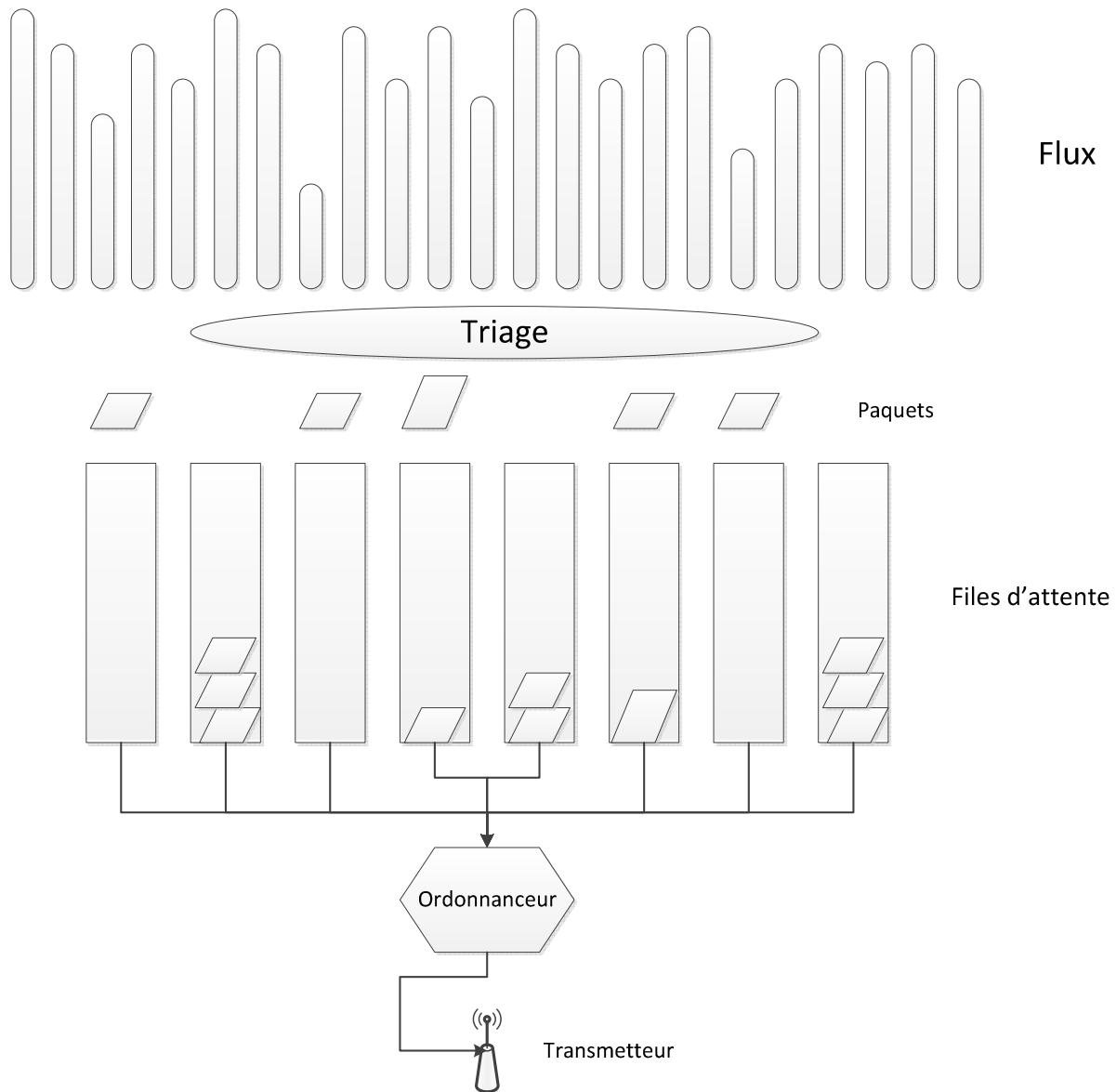


Figure 3 triage des flux et files d'attente

Les algorithmes de sceau percés représentent une façon simple et efficace de gérer la congestion dans les réseaux commutés par cellules comme ATM (*Asynchronous Transfer Mode*) et les autres réseaux à large bande commutés par paquets comme POS (*Packet Over SONET*) et Ethernet. En effet, ces algorithmes permettent de déterminer si un flux ou un ensemble de flux de communication dépassent une allocation de bande passante dans un temps donné.

L'efficacité de ces algorithmes a été étudiée dans [23] sur les marqueurs colorés, dans le contexte d'une simulation de réseaux congestionnés. C'est la technologie trTCM (*two rate Three Color Marker*) qui est utilisée dans ces travaux. Cette technologie avancée a été proposée initialement dans le cadre des conférences de L'IETF (RFC 2698) afin d'assurer la qualité du service dans les réseaux commutés par cellules ou par paquets, c'est-à-dire sans réservation stricte de bande passante, autrement dit dans le type de télécommunication qui utilise le principe du meilleur effort. La technologie trTCM se définit comme suit « *two rate Three Color Marker* » c'est-à-dire « Marqueurs Trois Couleurs deux taux » en français. Les deux taux en question sont calculés en utilisant l'algorithme de seau percé. Un premier taux représente la valeur CIR, ou le taux d'information contracté et un deuxième taux représente la valeur PIR, ou le taux d'information de pointe. Les valeurs CBS et PBS (*Committed Burst Size et Peak Burst Size*) sont aussi utilisées dans la configuration pour déterminer la taille des seaux en plus de leurs débits (taux). Donc un flux ou un ensemble de flux sont confrontés à ces valeurs de taux en utilisant l'algorithme de seau percé afin d'assigner une couleur aux cellules ou aux paquets admis. Ces couleurs permettront de décider quelles cellules ou quels paquets devront être rejetés, soit à l'admission ou à la transmission dépendant de l'endroit du traitement ou ceux-ci auront été colorés. Précisément, si le taux des cellules ou paquets du flux ou de l'ensemble de flux sont en deçà de la valeur de bande passante (synonyme du taux d'information) contracté (CIR, CBS), ceux-ci seront marqués de la couleur verte. Si par contre cette valeur est dépassée, mais que la valeur de pointe (PIR, PBS) n'est pas dépassée, les cellules ou paquets seront marqués de la couleur jaune. Finalement, si la valeur de pointe est dépassée aussi, les paquets seront marqués de la couleur rouge. Par la suite une fois les cellules ou les paquets marqués par ces couleurs par le système de classification du commutateur, ceux-ci seront prêts à être insérés dans les files d'attente pour traitement ultérieur, pourvu qu'ils remplissent les conditions de l'algorithme WRED (*Weighed Random Early Discard*) par exemple, si cette technologie est utilisée. L'efficacité de la technologie avancée WRED a été caractérisée dans l'article [24]. Cet algorithme utilise des courbes de probabilité de rejet pour chaque couleur

selon la quantité de cellules ou paquets en file d'attente afin de décider s'il y aura admission ou rejet. Normalement les courbes de probabilité de rejet sont linéaires ou même exponentielles avec la quantité de données dans les files d'attente, avec des pentes douces pour la couleur verte, moyennement abrupte pour la couleur jaune et très abrupte pour la couleur rouge. En combinant les technologies trTCM et WRED, les flux et les ensembles de flux de communication peuvent être comparés à des taux d'admission contractés et de pointes et ainsi gérer la bande passante de façon équitable en augmentant la probabilité de rejet quand les taux sont dépassés et la congestion existe.

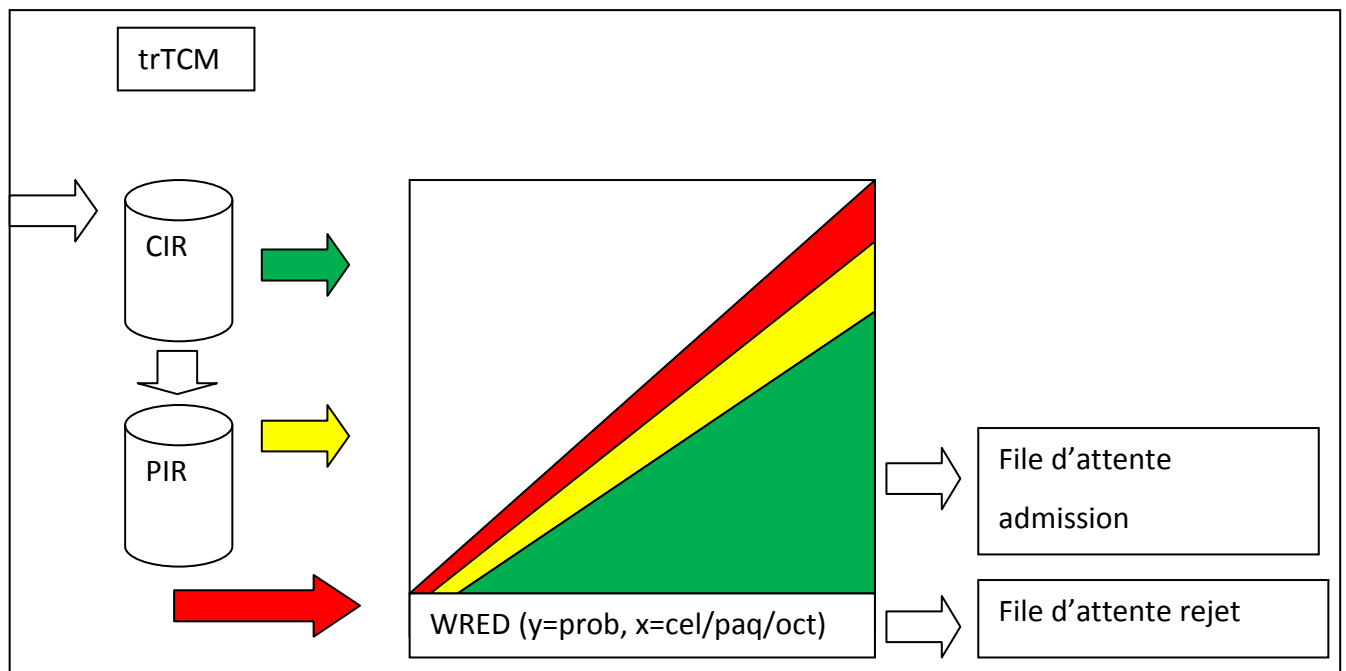


Figure 4 marqueurs trois couleurs deux taux

L'état de l'art sur les techniques de gestion des files d'attente a été présenté au niveau des couches basses même si elles ne représentent pas des couches comme telles parce qu'elles font partie des algorithmes qui constituent la fonctionnalité des couches basses.

3.2 Couche réseau

Pour l'analyse de l'état de l'art relativement à la couche réseau – et du routage, nous considérons deux sections. La première section décrit l'adressage conditionnel, et la deuxième section décrit le routage sans fil.

3.2.1 Adressage conditionnel

Dans les articles [7] et [25], on décrit des protocoles de routage de paquets dans un réseau VANET au niveau de la couche réseau du modèle OSI, et on caractérise la performance du protocole par simulation. Cette approche nouvelle utilise des conditions pour le routage de paquets entre les véhicules. Ces conditions sont des relations positionnelles entre les véhicules ou les endroits du réseau routier. Par exemple, une condition de routage pourrait adresser les véhicules sur une route qui sont jusqu'à 30 km derrière un véhicule. Les conditions comme telles ne sont pas décrites dans les détails, mais la performance est comparée avec celles de quelques autres approches de routage (par exemple AODV, OLSR, fast OLSR, et HOP). La contribution importante se retrouve dans le cadre référentiel pour les concepts d'adressage de véhicules, parce que l'utilisation d'adresses uniques au véhicule n'est pas vraiment faisable, mais les relations entre les véhicules et les relations entre les véhicules et leurs localisations dynamiques sur le réseau routier doivent être exploitées en tant que forme d'adressage. Le concept clé des articles, soit les conditions ou les relations de routage, sont trop peu considérés dans ces travaux, pour plutôt mettre l'accent sur la performance de l'approche dans quelques situations simulées. Il faudrait aller plus loin dans l'expérimentation de cette approche, l'actualiser et y appliquer des contraintes de qualité de service. Certaines hypothèses définies dans ces travaux omettent de considérer des critères importants. Par exemple on utilise l'hypothèse que les véhicules arrêtés n'ont pas la même trajectoire que les

véhicules en mouvement. Ceci pourrait être faux dans les situations de trafic routier, car des véhicules arrêtés pourraient être en avant de véhicules en mouvement et avoir exactement la même trajectoire, mais être arrêtés simplement parce qu'ils sont dans un bouchon de circulation.

La machine à états finis illustrée en Figure 5 automate de décision d'évitement de bouchon, montre la conception de la machine à états qui permet la décision de contournement et de détection de bouchon. Ce mécanisme représente le même processus de décision que ferait un conducteur de véhicule seulement avec l'information visuelle classique. Un conducteur approchant un bouchon situé à proximité d'une sortie de route pourrait prendre la décision d'effectuer la sortie et ainsi contourner le bouchon. Dans ce cas, le protocole de communication du système de transport intelligent pourrait étendre le champ de vision virtuel du conducteur, en trouvant des bouchons et des possibilités de sorties qui ne se trouvent pas dans de champ de vision réel du conducteur. De plus, un ordinateur de bord associé au système de communication pourrait servir de navigateur virtuel en calculant l'impact du bouchon sur le temps de parcours ou la consommation d'essence par rapport à l'alternative de contournement.

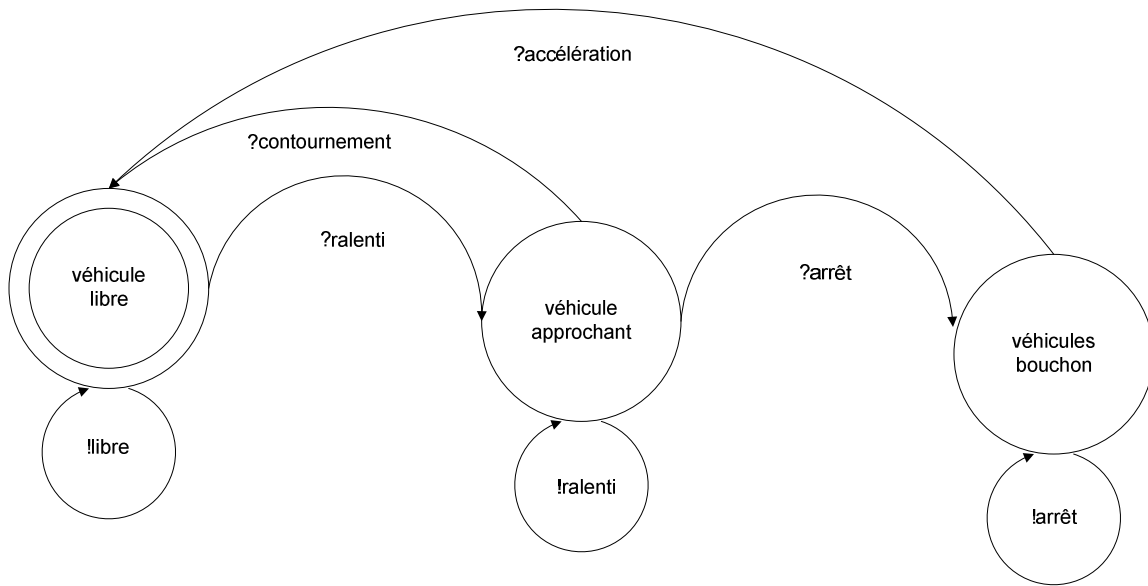


Figure 5 automate de décision d'évitement de bouchon

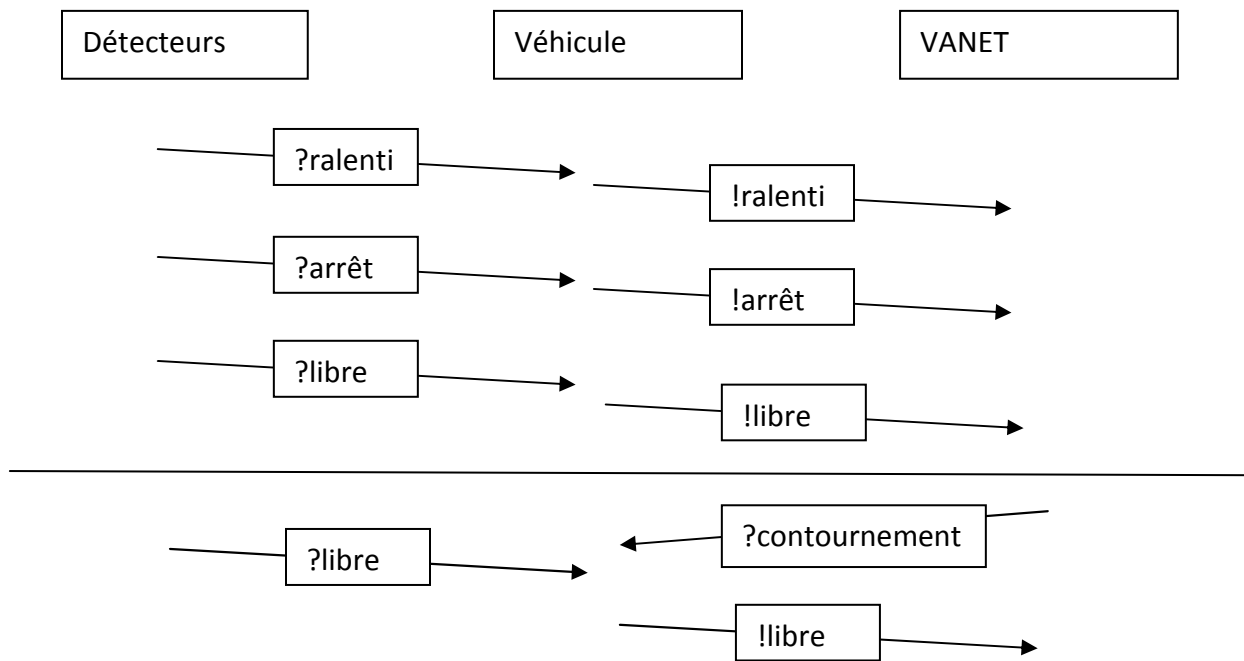


Figure 6 diagramme de séquence d'évitement de bouchon

3.2.2 Routage des réseaux ad hoc

Les travaux présentés dans [6] décrivent un protocole de routage de paquets pour découvrir le parcours routier le plus rapide dans des situations de trafic véhiculaire. Le protocole décrit des algorithmes de « drop » de paquets, de détour de régions vides de cellules véhiculaires et de guidage en temps réel. Cette contribution répond au besoin d'un algorithme de routage qui tient compte des informations qui peuvent être mise à disposition en temps réel par les réseaux véhiculaires sans-fil ad hoc. La problématique soulevée est que les paquets qui contiennent l'information de routage en temps réel ne pourront pas nécessairement prendre le plus court chemin, parce qu'il est possible qu'il y ait des trous dans le chemin, et bien sûr il est possible que le plus court chemin ne soit pas le plus rapide parce que le trafic se trouve ralenti pour diverse raison sur ce chemin. La simulation a été utilisée dans l'expérience pour démontrer l'efficacité de l'algorithme. Les trois algorithmes de « drop » de paquets, long drop, loop

duplication drop, et convergence duplication drop ne sont pas décrits dans les détails. Si les « drop » ne sont pas faits de façon efficace, le surplus de paquets sur le réseau pourrait causer de la congestion inutile, et les paquets non « droppés » pourraient n'avoir aucune utilité sur le réseau. Des protocoles existants auraient dû être référés pour régler ces problèmes de « drop », ou l'article aurait dû expliquer pourquoi les détails sont omis. Dans ce contexte, les cellules sont définies comme étant un regroupement de véhicules sur le réseau de communication. Les segments sont aussi définis comme des parcelles du réseau routier entre deux intersections. L'article semble tenir pour acquis que les cellules gardent un historique des paquets et qu'elles utilisent cet historique pour prendre les décisions de « drop ». Un tel historique pourrait avoir besoin d'énormément de mémoire, et d'algorithmes de recherche coûteux, ce qui n'est pas pratique pour les applications véhiculaires qui sont limitées en mémoire et en puissance de traitement. Les algorithmes de détour et de guidage sont intéressants, mais ils n'ont pas été mis ensemble au cours de l'expérimentation, et les effets des algorithmes d'échappement non plus. De plus la situation ou la destination routière tombe dans une cellule inexistante (segment vide de cellules), n'est pas décrite comme cas particulier. Donc le protocole décrit dans ces travaux manque de détails d'implémentation qui sont pertinents. Par exemple, les décisions de routage devraient être prises au niveau des cellules véhiculaires pour réduire la quantité de données que pourraient représenter l'historique des paquets, il faudrait expérimenter avec ce protocole pour vérifier cette hypothèse.

3.3 Couches supérieures

L'analyse de l'état de l'art pour les couches supérieures est divisée en deux sous-sections, soit une sous-section pour la couche TCP et une sous-section pour la couche application.

3.3.1 Couche TCP

Le contrôle de flux effectué à la couche niveau 4 des protocoles TCP/IP est en effet un bon exemple des efforts qui sont investis pour tenter d'éviter les problèmes causés par les situations de congestion. Le protocole TCP, qui est orienté connexion et ainsi requière des confirmations de réceptions, s'adapte aux paquets manquants. Pour TCP, il n'est pas possible de savoir si les paquets sont manquants à cause des erreurs de transmission ou parce qu'il y a de la congestion, mais le comportement du protocole est adapté aux situations de congestion parce que c'est à ce niveau qu'il peut agir, alors qu'il ne peut pas vraiment faire quoi que ce soit pour les erreurs de transmission. Quand plusieurs conversations TCP ont lieu en même temps, et que le lien physique et les ressources y donnant accès sont congestionnés, les paquets perdus entraînent les mécanismes du contrôle de flux qui ralentissent les conversations. Dans les scénarios pathétiques de congestion, un problème de synchronisation globale peut se produire, ce qui représente un des pires cas de congestion [26]. Des efforts sont faits aussi en recherche au niveau des applications qui utilisent des protocoles de transports sans connexions comme UDP, RTP, SCTP pour éviter les situations de congestion. Par exemple les protocoles au niveau d'application pour la voix et la vidéo vont utiliser des techniques pour amoindrir les effets des paquets perdus sur le son et l'image [27, 28]. De plus, ces protocoles seront configurés dans plusieurs cas pour utiliser des niveaux de service avec meilleure priorité et ainsi profiter d'une meilleure qualité de service donc moins de perte de paquets. Les protocoles de routage qui sont impliqués à ce niveau offriront la possibilité de configurer la classification des paquets afin d'offrir la bonne qualité de service aux applications qui en ont besoin. De plus ces protocoles offriront des possibilités de redondance des liens et du balancement de charge pour fournir une qualité de service adéquate.

3.3.2 Couche application

Une fois l'infrastructure en place pour les réseaux de transports intelligents, les applications possibles sont infinies. La gamme d'applications de divertissement déjà disponible sur les

ordinateurs personnels et les appareils de communications mobiles deviennent aussi disponibles dans les véhicules. Les passagers ont accès à une bonne partie de la bande passante pour leur divertissement. Les applications permettant le contournement de bouchon de circulation deviendront disponibles comme dans notre exemple de scénario.

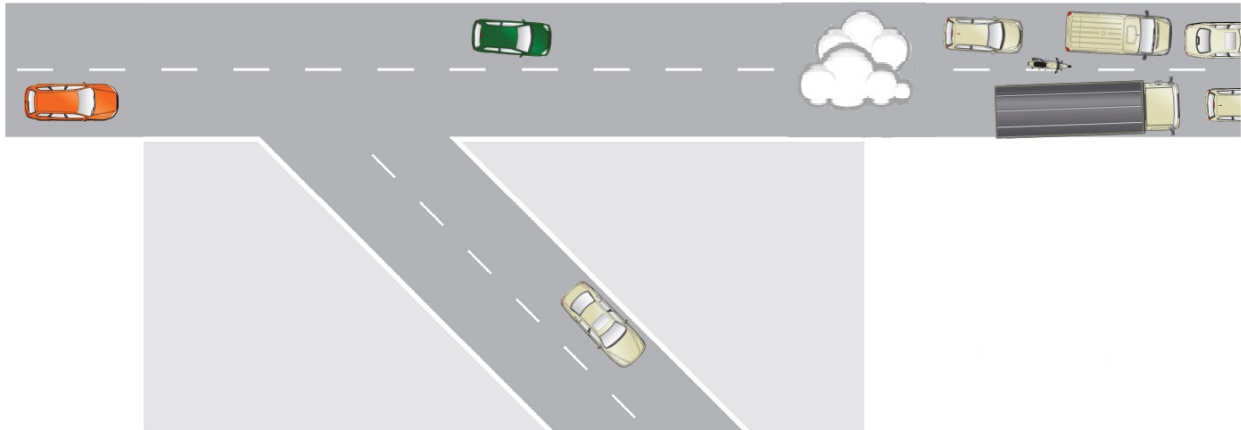


Figure 7 application de contournement

Par exemple dans l'article de guidage en temps réel [29], on retrouve un algorithme qui permet de choisir le meilleur parcours dans un réseau routier en utilisant le réseau de communication sans fil. Dans la Figure 7 application de contournement on retrouve notre scénario exposé en problématique. Le véhicule à gauche ne peut pas savoir qu'il y a un bouchon après la prochaine sortie en utilisant seulement l'information visuelle, alors que l'application de contournement peut l'informer du bouchon afin qu'il puisse prendre la sortie à temps.

3.4 Concepts d'architecture intercouches

La section qui suit présente deux sous-sections pour les concepts d'architectures qui sont basés sur des concepts innovateurs et qui se caractérisent par des communications inter couches et qui peuvent tirer avantage des techniques particulières comme l'intelligence artificielle.

3.4.1 Intelligence artificielle

Dans l'article [30], on propose d'utiliser les fourmis comme agents pour le routage dans les réseaux mobiles ad hoc. Les solutions basées sur des espèces naturelles d'intelligence simple comme les fourmis sont des techniques d'intelligence artificielle. Ces fourmis font partie d'un cadre de travail d'optimisation appelé ACO (*Ant Colony Optimization*, optimisation de colonies de fourmis) inspiré des colonies de fourmis qui existent dans la nature. Les colonies de fourmis dans la nature utilisent les phéromones qu'elles émettent et perçoivent pour évaluer la qualité d'un chemin, le chemin le plus court ayant une meilleure qualité. L'aspect hybride du routage vient du fait que l'algorithme proposé est à la fois réactif et proactif. L'algorithme consiste donc à distribuer les fourmis et les données au besoin, d'où l'aspect réactif, sur les chemins différents de façon stochastique. Par la suite, après avoir reçu une rétroaction venant des fourmis qui reviennent, de choisir les chemins préférables selon leur qualité estimée et de continuer d'envoyer des fourmis pour évaluer la qualité des chemins choisis, d'où l'aspect proactif. L'expérience de l'article compare avantageusement l'algorithme *AnthocNet* à l'algorithme AODV dans un scénario de simulation vient du fait que d'après eux les autres expériences n'ont pas intégré tous les ingrédients nécessaires pour avoir une solution complète. L'expérience présentée fait des compromis entre les systèmes d'intelligence artificielle et les systèmes classiques, en mélangeant les fourmis et les hellos en plus de l'approche hybride avec le proactif et le réactif et la distribution stochastique qui fournit le balancement de charge. C'est intéressant d'avoir des approches hybrides, mais une méthode plus puriste aurait probablement été plus convaincante, et les problèmes d'AODV à la base ne sont pas vraiment exposés, à part de dire qu'AODV est simplement réactif. De façon pragmatique, une fourmi c'est un hello et un hello c'est une fourmi, c'est leurs comportements qui soit relèvent de l'intelligence artificielle ou d'une algorithmique bête, c'est là-dessus qu'on aurait du mettre l'emphase pour cette expérience. L'aspect intéressant des fourmis qui n'a pas été exploré dans cet article, est que celles-ci trouveront des chemins supplémentaires dans les cas de congestion, mais on a tout simplement évité le sujet en limitant les fourmis. Nous

croyons qu'une meilleure méthode à adopter rendrait les fourmis et les données une seule et même entité, et approfondir la recherche dans cette direction serait une meilleure contribution scientifique.

Dans l'article [31], on propose un cadre de travail où la programmation autonome autogèrerait des réseaux sans-fils hybrides. Ces réseaux cellulaires seraient composés d'un mélange de réseau sans fil avec infrastructure et sans infrastructure, ou ad hoc, d'où l'appellation hybride. Ces nouvelles architectures de réseaux cellulaires seraient nécessaires vu la croissance des besoins en communication et ils amèneraient avec eux une problématique supplémentaire de gestion. Le concept de programmation autonome ou communications autonome dans le contexte des réseaux permettrait d'alléger la tâche de gestion des réseaux sous quatre aspects différents. L'autoconfiguration, l'auto-optimisation, l'auto-réparation, et l'auto-protection. L'architecture des systèmes autonomes de communication comprend une collection d'éléments autonomes interactifs couplés d'un gestionnaire autonome qui les supervise et les gère, à chaque couche de l'architecture des réseaux implémentée. Les processus basés sur les concepts empruntés de l'intelligence artificielle comme la supervision, l'analyse, la planification et l'exécution sont sous la responsabilité du gestionnaire autonome. Ces gestionnaires pourraient aider à minimiser les interventions manuelles sur les réseaux. L'article donne pour exemple la gestion par politiques d'un réseau comme une application de l'architecture autonome. Les politiques seraient exprimés selon leurs types, soit action, but ou utilités allant du plus bas niveau pour les politiques de type action au plus haut niveau pour les politiques de type utilités. L'agent de gestion autonome utiliserait une approche d'apprentissage machine comme les réseaux de neurones pour intervenir selon les différentes politiques en trois étapes, la première étant l'analyse de l'état courant à savoir si les buts sont atteints, la seconde étant l'utilisation d'heuristiques pour trouver les politiques pertinentes et la troisième étant la prise de décision selon les politiques pertinentes trouvées. Le second exemple donné est plus concret, il s'agit de l'allocation dynamique distribuée des canaux de niveau MAC, où les nœuds peuvent faire

l'allocation des canaux en utilisant un réseau de neurones et l'apprentissage machine et ainsi éviter la complexité d'une gestion centralisée. Malheureusement, le cadre de travail et les exemples sont les seules contributions dans cet article, il n'y a pas d'expérience sous simulation ou même de données prises sur le terrain.

Dans l'article [32], on propose une solution de routage pour optimiser les communications dans les réseaux sans fil ad hoc et ainsi les rendre vertes, ou écologiques. Selon eux les communications sans fil seraient polluantes de deux façons, en générant des ondes radio et en drainant leurs sources d'énergie. Les communications inefficaces qui sont victimes d'interférence rendraient cette pollution encore pire. Le routage intelligent pourrait être une solution à ce problème de pollution en onde radio et en énergie. Les radios cognitives décrites précédemment seraient un lieu de prédilection pour le routage intelligent, les appareils mobiles pourraient apprendre en utilisant les stimuli environnants pour prendre des décisions de routage au sein du réseau ad hoc qui économiserait les ondes radio et l'énergie requise pour les produire. On a utilisé un système de simulation nommé EnVO, implémenté sur le système Matlab pour tester l'approche du routage intelligent à différents niveaux avec trois algorithmes de routage, le routage inondant (*flood routing*), le routage LAR (*Location Aware Routing*, routage tenant compte de la localisation) et le routage SLAM (*Simultaneous Location And Mapping*, routage tenant compte de la localisation et de la topologie). Les résultats obtenus en simulation sont intéressants pour caractériser les avantages du routage intelligent versus le routage inondant. Les avantages se voient au niveau de la performance et de l'économie en interférence d'ondes radio dans leur rapport. Par contre on n'a pas caractérisé l'économie en énergie, probablement que cette donnée ne faisait pas partie du modèle, c'est malheureux parce qu'en introduction ceux-ci ont mis beaucoup d'emphasis sur l'importance de l'économie d'énergie. Dans l'article on a aussi expliqué les paramètres énergétiques utilisés dans la simulation, mais aucun résultat associé n'apparaît dans cet article.

Dans l'article [33], on discute des défis qu'amène la gestion autonome des réseaux. D'après eux, la gestion autonome deviendrait nécessaire vu la complexité grandissante des réseaux, et de la place envahissante que les appareils électroniques réseautés prennent dans nos vies. Ces appareils devraient dans le futur, contenir des logiciels intelligents qui seront capables de communiquer spontanément avec les appareils dans leur environnement proche, faire un monitoring constant des ressources et se reconfigurer de façon autonome afin de pouvoir garder leurs paramètres à l'intérieur des bornes définies. L'article présente les problématiques associées au design, à l'implémentation et au déploiement des réseaux autonomes, en focalisant sur l'aspect de gestion de ces réseaux. Par la suite on motive l'émergence des architectures de réseau autonome et expose quelques architectures qui ont été proposées à date. Ils discutent de la complexité associée à la modélisation de l'information et du comportement en contexte autonome, et de l'inspiration qui provient des modèles biologiques. En effet la programmation autonome est une branche de l'intelligence artificielle qui se veut calquée sur les comportements autonomes des systèmes biologiques. Les requis fonctionnels en gestion des systèmes autonomes sont l'auto localisation, l'auto configuration, l'auto réparation, l'auto optimisation et l'auto protection. Les architectures survolées sont les réseaux cognitifs DARPA (*Defense Advanced Research Projects Agency*), l'architecture bioréseau et l'architecture FOCAL (*Foundation, Observation, Comparison, Action, and Learning Environment*). Finalement l'article ne propose pas de solution propre à un contexte, mais il s'agit bien d'un état de l'art sur l'utilisation de la gestion autonome pour l'évolution des réseaux du futur.

Dans l'article [34], on propose une solution pour gérer les réseaux maillés sans fils cognitifs en utilisant l'apprentissage par renforcement. On suggère que les solutions actuelles de routage pour les réseaux sans fil d'appareils mobiles manquent d'informations contextuelles des conditions du réseau pour bien optimiser l'utilisation des ressources limitées de ces appareils, notamment les capacités de batteries relativement petites. Ces réseaux apportent leur lot de problème d'optimisation, dont la longueur des chemins de routage, le balancement des

charges, la gestion consistante des liens et l'agrégation. La plupart des solutions de routage existantes optimisent un de ces aspects, mais quand vient le temps d'optimiser l'ensemble de ceux-ci, des techniques comme l'apprentissage par renforcement deviennent nécessaires. De plus, on propose l'architecture d'apprentissage par renforcement nommée apprentissage-Q qui utilise une approche intercouche qui permet aux agents de réseautage autonome de s'adapter, s'autoconfigurer et s'autogérer, et ce de façon optimale. Ils ont implémenté un prototype basé sur le protocole AODV et on démontré qu'avec les contraintes des usagers, soit la qualité de service (QoS) et les ententes de service (SLA) définies, leur système simulé sous OPNET optimise bien l'utilisation des ressources. Le point qui rend ce travail de recherche moins intéressant est le fait que l'utilisation de l'apprentissage par renforcement est justifiée en disant que les protocoles existants n'optimisent qu'un aspect du routage, alors qu'avec cette solution expérimentale, c'est exactement la même chose qui est accomplie, on ne travaille que sur un seul paramètre du routage, soit la longueur des chemins.

3.4.2 Techniques intercouches

Dans l'article [35], on donne un tutoriel sur l'optimisation des ressources pour les réseaux sans fil en utilisant des techniques intercouches (cross-layer). La problématique explorée en est une d'optimisation des ressources dans le contexte des communications sans fil. On a tout d'abord observé les phénomènes d'optimisation d'une façon myope, c'est-à-dire en limitant le problème en utilisant un modèle d'ordonnancement opportuniste avec un seul saut, avec l'emphase sur l'intégration de l'ordonnancement avec les informations de la couche physique. Par la suite ils ont ajouté au problème le contrôle de congestion et finalement ils ont proposé une solution plus élégante avec une approche intercouche. La solution finale proposée utilise une méthode d'ordonnancement imparfait, vu la complexité des algorithmes à coder et leur gourmandise en ressources, mais celle-ci a quand même la possibilité d'exprimer des politiques permettant de fournir une bonne qualité de service et l'équité des flux de communication. La démonstration de la validité de l'approche suggérée est faite de façon formelle, soit purement mathématique. On suggère que les formules fournies par leurs recherches pourraient être une

base pour une implémentation nouvelle, partie de zéro, avec une approche intercouche. Les formules proposées sont en fait difficiles à déchiffrer pour un chercheur qui est plus habitué aux preuves par simulation qu'aux preuves mathématiques. De plus on expose des manquements aux modèles mathématiques en proposant d'y travailler dans le futur, ce qui veut dire que les formules applicables seraient probablement encore plus complexes.

3.5 Synthèse

Pour gérer la congestion dans un contexte comme celui des réseaux sans fil ad hoc de véhicules, où on ne peut pas prédire que des messages seront plus importants que d'autres pour les flux de communication de type meilleur effort, l'état de l'art nous fournit des pistes de solution. On a vu dans le chapitre de la problématique qu'on fait l'hypothèse que l'équité serait une bonne méthode pour optimiser la qualité de service des flux de communication. Pour que cette hypothèse se tienne, il faudrait simplement que les flux d'information qui permettent la gestion du trafic routier tirent avantage de l'équité en utilisant explicitement moins que leur juste part de bande passante ce qui leur mériterait des meilleurs délais, sans avoir besoin d'utiliser la priorité de la qualité de service ou les canaux d'urgence. On peut ajouter ici l'hypothèse que les applications de gestion du trafic routier tireront avantage de l'équité fournie comme caractéristique de la QoS du réseau sans fil ad hoc. Partant de cette hypothèse il faut donc travailler sur les couches de réseau inférieures à la couche application pour implémenter un système qui fait adéquatement la promotion de l'équité. Pour ce faire, il faudrait intégrer les mécanismes de coordinations tels l'EDCF [3], les algorithmes de gestion de files d'attente hybrides approximant WFQ [22], le routage par transmissions conditionnelles [7] et l'intelligence artificielle [34] dans une approche intercouche [35] comme vue précédemment dans l'état de l'art. Ce système de communication VANET, avec les modifications des algorithmes identifiés à partir de notre revue de la littérature, qu'on proposera dans le chapitre de la méthode, pourrait optimiser la qualité de service pour diverses applications, dont l'application potentielle de gestion du trafic routier.

La méthode EDCF comme algorithme à étudier pour les couches basses se justifie par le fait que la spécification standard de la DCF des protocoles de la famille 802.11 ouvre la porte à ce genre de technologie avec la fenêtre de contention déterminée avec un paramètre temps aléatoire. La fenêtre de contention permet à un nombre de stations radio en état de collision de se retirer pendant un certain temps afin que la prochaine trame envoyée par ces stations ne provoque pas une collision à nouveau. Cette fenêtre est composée d'un temps de retrait qui grandit de façon binaire exponentielle auquel on ajoute un paramètre de temps tiré au hasard. En effet, le paramètre temps tiré du hasard peut être un paramètre qui contient une composante qui influence la qualité de service, ce qui semble provenir moins du hasard à la base. Si on utilise l'hypothèse que la distribution des classes de trafic répond aux règles du hasard pour un réseau véhiculaire ad hoc donné, la composante du hasard existe de toute façon dans le calcul de la fenêtre de contention. D'ailleurs la technologie de bas niveau IDMA comparée à CDMA utilise une approche semblable pour la détermination des symboles, car ceux-ci proviennent d'un hasard contrôlé pour améliorer la distribution des symboles.

Pour les algorithmes à étudier en ce qui concerne l'ordonnement des files d'attente, il faut utiliser un algorithme qui permet d'approximer les algorithmes idéaux comme WFQ et ses variantes. La flexibilité de l'algorithme WFQ apporte une complexité [16] qui n'est pas pratique dans le contexte qui nous intéresse, alors un algorithme qui simplifie les paramètres utilisés pour trier les flux et les files d'attente comme TSFQ est tout à fait approprié. En effet c'est surtout le tri qui apporte la complexité de traitement dans le cas des algorithmes de files d'attente qui modélisent l'équité, mais quand on se rend compte du contexte comme celui de la QoS appliqué aux réseaux véhiculaires, la simplification des paramètres et la hiérarchisation des classes permettent de sauver du temps de traitement.

Pour l'algorithme de routage au niveau de la couche transport, les transmissions conditionnelles sont tout à fait adéquates. L'adressage anonyme des transmissions conditionnelles permet aux véhicules de communiquer dans le contexte d'échange

d'information sur l'état du trafic routier sans avoir recours aux méthodes d'adressage conventionnel qui est pris pour acquis dans les réseaux filaires. Dans ce modèle, tous les véhicules sont potentiellement des routeurs, et les algorithmes d'élection de routeurs désignés comme les technologies à détermination de convois sont éliminés. Il est possible que ce modèle engendre une plus grande utilisation de la bande passante, et il ne serait donc possiblement pas optimal, mais pour notre contexte d'expérimentation sur l'équité et la qualité de service dans des cas de congestion du réseau dans les réseaux hybrides où l'infrastructure est souvent inexistante, il n'existe pas d'alternatives viables. L'approche de routage avec formation de convois dynamiques [36] requiert la compatibilité des trajectoires, alors qu'ici on introduit la possibilité de contournement, comme vu dans notre scénario en problématique. Le tableau qui suit résume la sélection de technologies à étudier.

	Technologie retenue	Technologies non retenues
Couches basses	EDCF	CDMA/IDMA
<i>Critères</i>	<i>Flexibilité avec le hasard de la fenêtre de contention</i>	<i>Rigidité au niveau du hasard prouvé comme étant meilleur</i>
Files d'attente	TSFQ	DRR/WFQ
<i>Critères</i>	<i>Adéquation du modèle avec les limites du contexte VANET</i>	<i>Trop simple et inéquitable/ Trop flexible pour le contexte</i>
Routage	Transmissions conditionnelles	Convois dynamiques
<i>Critères</i>	<i>Dynamacité des décisions de routage par véhicules</i>	<i>Complexité du processus d'élection et de convoyage</i>

Figure 8 choix des technologies

Nous allons expérimenter cette approche qui intègre les trois technologies choisies dans les chapitres suivants, après avoir fait la synthèse des trois algorithmes qu'on va utiliser comme base dans notre solution.

3.5.1 Algorithme EDCF

L'algorithme de l'EDCF est une amélioration de l'algorithme DCF parce qu'il tient compte de la priorité des classes de service auquel le paquet en attente de transmission fait partie. La priorité des classes de service est le paramètre de base du modèle des services différenciés de qualité de service. Cette priorité peut venir des paramètres configurés de qualité de service, ou de l'élévation de priorité accordée à un flux pour promouvoir l'équité. L'article [3] décrit cet algorithme en spécifiant des tables pour les valeurs d'AIFS (*Arbitration Inter Frame Space*), CW_min et CW_max (*Contention Window*). Ces valeurs sont les composantes de la fenêtre de retrait BEB décrite dans la Figure 1 schématisation des délais de l'EDCF. Les tables prévoient quatre classes de service. L'algorithme est celui de la boucle de transmission avec les fonctions d'initialisation de la valeur de retrait et d'augmentation de la valeur de retrait. Le code qui suit ne contient pas tout l'algorithme de la boucle de transmission, mais seulement la partie qui est pertinente à la fonction EDCF. La priorité d'un paquet est augmentée quand le délai de retrait qui lui est accordé selon sa classe est plus petit. Conséquemment la priorité d'un paquet est diminuée quand le délai de retrait qui lui est accordé selon sa classe est plus grand.

L'algorithme EDCF calcule une valeur de temps d'attente avant de transmettre sur le réseau sans fil à partir de la priorité, du nombre de collisions précédentes et une variable tirée de manière aléatoire.

```
// tableau des Adjusted Inter Frame Space
// en unite de temps d'octets

const int AIFS [] = { 2, 3, 6, 9 }

// tableau des fenetres de contention minimales
// Contention Window en unite de temps d'octets
```

```

const int CW_min [] = { 3, 7, 15, 15 }

// tableau des fenetres de contention maximales
// Contention Window en unite de temps d'octets

const int CW_max[] = { 7, 15, 1023, 1023 }

// init_retrait: fonction d'initialization des retraits
// parametres:
// classe: classe de service
// resultat: valeur initiale du retrait pour une classe donnee

int init_retrait ( int classe )
{
    return ( CW_min [ classe ] + AIFS [ classe ] )
}

// augmenter_retrait: fonction d'augmentation des retraits
// parametres:
// classe: classe de service
// retrait: retrait en unite de temps d'octets
// resultat: valeur augmentee du retrait pour une classe donnee

int augmenter_retrait ( int classe, int retrait )
{
    retrait = retrait - AIFS [ classe ]
    retrait = retrait * 2
    retrait = retrait + 1

    if ( retrait > CW_max [ classe ] )
    {
        retrait = CW_max [ classe ]
    }

    return ( retrait + AIFS [ classe ] )
}

// transmission: fonction de transmission de paquets decrite
// minimalement ici avec l'utilisation d'une structure de
// donnee abstraite paquet qui contient un chant retrait de
// type int et une variable collision de type int aussi

```

```

// la fonction de transmission n'est pas decrite completement
// afin d'allegger le texte

void transmission ( void )
{
    ...

    paquet->retrait = init_retrait ( classe )

    ...

    if ( collision )

        paquet.retrait = augmenter_retrait ( classe, retrait )

    sinon

        paquet.retrait = initialiser_retrait ( classe )

    ...

    transmettre ( paquet )

    ...
}

```

Figure 9 algorithme EDCF

3.5.2 Algorithme hybride de gestion des files d'attente

Les algorithmes hybrides de gestion des files d'attente sont des algorithmes qui font un compromis entre le modèle idéal d'équité WFQ et la simplicité du modèle couramment utilisé DRR. Par exemple l'algorithme décrit dans [22], combine l'utilisation des paramètres de taille de paquets utilisés dans DRR pour modéliser le temps virtuel d'achèvement de transmission et le tri des flux de WFQ en limitant le nombre d'entrées et en hiérarchisant la structure des files d'attente. Pour ce faire, l'article prouve que la simplification des calculs et les limitations permettent de borner les délais et garder la complexité de calcul en $O(\log(n))$. L'algorithme qui

suit représente une implémentation de cet algorithme en faisant abstraction des fonctions de simplification parce qu'elles dépendent du nombre borné de flux et de paquets du système.

L'algorithme TSFQ permet d'ordonner le vidage des files d'attente selon la priorité des flux tout en permettant de borner les délais pour une certaine équité.

```
// comparer_flux: fonction qui permet de comparer deux flux
// cette fonction est concue pour etre appelee par la fonction
// de librairie standard qsort pour faire un tri rapide
// parametres:
// flux_A: pointeur sur un premier flux
// flux_B: pointeur sur un second flux

int comparer_flux ( struct flux_t *flux_A,
                   struct flux_t *flux_B )
{
    // variables servant a quantifier les priorites de flux
    int quant_A, quant_B

    // les fonctions simplifier_* sont omises afin d'alléger le
    // le texte, elles permettent de quantifier les priorites
    // selon leur parametre specifique et des variables globales
    // ajustees a chaque iteration afin de borner les valeurs
    // possibles

    quant_A = simplifier_temps ( flux_A.paquet_tete.arrivee )
             + simplifier_taille ( flux_A.paquet_tete.taille )
             + simplifier_taille_totale ( flux_A.taille_totale )
             + flux_A -> classe

    quant_B = simplifier_temps ( flux_B.paquet_tete.arrivee )
             + simplifier_taille ( flux_B.paquet_tete.taille )
             + simplifier_taille_totale ( flux_B.taille_totale )
             + flux_B -> classe

    return ( quant_A - quant_B )
}

// traitement: fonction de traitement des paquets decrite de
// facon minimale afin d'alléger le texte, elle fait appel
// aux fonctions trier_flux, extraire_paquet et transmettre
// qui sont omises afin d'alléger le texte aussi, la description
```

```

// des structures de donnees packet et vecteur_flux sont omises
// pour les memes raisons

int traitement ( void )
{
    ...

    // la fonction trier standard prend la fonction de
    // comparaison en parametre, cette fonction est appelee
    // par la fonction trier_flux qui suit

    trier_flux ( vecteur_flux, comparer_flux )

    ...

    packet = extraire_packet ( vecteur_flux )

    ...

    transmettre ( packet )

    ...
}

```

Figure 10 algorithme hybride de gestion des files d'attente

3.5.3 Algorithme transmissions conditionnelles

L'algorithme des transmissions conditionnelles permet de router les paquets entre les nœuds des réseaux ad hoc selon la pertinence des paquets dans leur contexte autoroutier. L'article [7] ne décrit pas l'algorithme comme tel, mais il expose les éléments de données qui composent la stratégie de routage. Les paquets de routage contiennent les paramètres de localisation, trajectoire et vitesse des véhicules transmetteurs et des conditions de transmission selon la localisation, trajectoire et vitesse des véhicules receveurs. L'algorithme qui suit, sans tenter d'exprimer les idées des transmissions conditionnelles de façon exhaustive, donne une idée d'une implémentation supportant les transmissions conditionnelles.

```

// verifier_condition: fonction qui verifie les conditions de

```

```

// transmission d'un paquet selon le protocole de routage
// dynamique des transmissions conditionnelles
// parametres:
// packet: pointeur sur le paquet a verifier
// resultat: entier representant une valeur booleene
// a savoir si la condition de transmission est remplie

int verifier_condition ( struct packet_t *packet )
{
    switch ( packet->condition.type )
    {
        case MEME_TRAJET :

            if ( comparer_vitesse ( paquet->condition.vitesse,
                                    VITESSE_NULLE ) )
                return ( FAUX )

            if ( comparer_trajet ( paquet->condition.trajet,
                                   vehicule.trajet ) )
                return ( FAUX )

            if ( comparer_distance ( paquet->condition.distance,
                                     vehicule.position)
                return ( FAUX )

            return ( VRAI )

        case TRAJET_DIFFERENT :

            ...

    }
}

// transmission: fonction de transmission de paquets décrite
// minimalement ici avec l'utilisation d'une structure de
// donnee abstraite paquet qui contient un champ retrait de
// type int et une variable collision de type int aussi
// la fonction de transmission n'est pas decrite completement
// afin d'alléger le texte

void transmission ( void )

```

```

{
    ...
    transmettable = verifier_condition ( packet )
    ...
    if ( transmettable )
        transmettre ( packet )
    else
        rejeter ( packet )
    ...
}

```

Figure 11 algorithme transmissions conditionnelles

Avec ces trois algorithmes et le concept des radios cognitives, nous avons une base de travail pour le chapitre suivant, soit la méthode. Dans ce chapitre, nous reprendrons ces trois algorithmes avec les modifications qui permettront un système intégré qui fait la promotion de l'équité pour optimiser la qualité de service et nous décrivons la démarche de validation de nos hypothèses.

Avec notre état de l'art, on a pu identifier des travaux qui traitent des situations de congestion dans les réseaux VANET et dans les réseaux filaires. Pour ces situations de congestion, on a pu voir que les concepts d'équité et de la qualité de service apportent des solutions pour gérer cette congestion. Par contre, il y a des lacunes à combler et c'est à ce niveau que nous situons nos contributions, qui sont basées sur des extensions des technologies de l'état de l'art dans le but de mieux répondre au besoin d'équité des communications entre les véhicules. La principale limite des travaux existants étant que dans l'état de l'art pour VANET, les solutions pour l'équité traitent au niveau des véhicules et non au niveau des flux. Une autre limite se

situé au niveau du routage par transmissions conditionnelles, qui est très adéquat pour le contexte VANET, mais considère les véhicules arrêtés comme n'ayant pas de trajectoires utiles.

4 Méthodologie

Pour notre méthodologie, nous apportons des solutions qui répondent aux objectifs fixés dans notre chapitre de la problématique. Nous procédons par la suite à leur développement et leur expérimentation pour pouvoir les valider et mesurer leur apports.

Rappelons que notre objectif principal est de développer une solution adéquate pour la gestion de la congestion des réseaux VANET, basé sur l'équité et la QoS. Nous précisons également que la solution que nous proposons se situe au niveau des flux de communication. Les objectifs secondaires que nous avons définis sont les suivants :

- Valider qu'on puisse définir un algorithme de routage dynamique en tenant compte de l'équité et de la QoS de façon intégrée.
- Valider qu'on puisse définir ce routage dynamique avec l'équité et la QoS intégrée de façon optimale.

Les sections suivantes discutent plus en détail de la réalisation de ces objectifs de notre recherche.

4.1 Objectif principal: L'équité et la QoS pour la gestion de la congestion des VANETs au niveau des flux de communication

Le traitement de l'équité, centré au niveau des véhicules, comme nous avons constaté dans l'étude de l'état de l'art est une lacune de la recherche sur les VANETs. Cette constatation nous apporte notre objectif principal tel que mentionné en problématique à savoir : définir un système qui va promouvoir l'équité et la QoS au niveau des flux dans un réseau sans fil ad hoc véhiculaire pour gérer la congestion. Il faut bien comprendre l'importance des flux de communication dans ce contexte, et que leur traitement se fait au niveau de la couche 4 du modèle TCP/IP. Les flux sont importants parce que ce sont les contenants d'un échange

complet entre les véhicules, et souvent les véhicules intermédiaires serviront seulement de relais pour ces messages, de routeurs du réseau ad-hoc en fait. Rappelons, comme il a été mentionné dans l'état de l'art, que l'équité est un concept qui se situe au niveau des flux de communication, et donc notre méthode devra le refléter en mesurant l'impact de nos modifications au niveau des flux plutôt qu'au niveau des véhicules. Pour ce faire, l'unité abstraite considérée pour la validation de notre méthode sera le flux de communication, et l'équité sera mesurée en comparant les délais moyens des différents flux de communication, en sachant que les flux qui utilisent moins que leur juste part de bande passante devraient se retrouver avec les meilleurs délais, alors que les flux qui utilisent plus que leur juste part de bande passante devraient se retrouver avec les pires délais et en s'assurant aussi que les délais sont bornés.

4.2 Objectif secondaire : Algorithme de routage dynamique intégrant l'équité et la QoS

Une autre lacune constatée dans le chapitre de l'état de l'art est que le lien entre l'équité, la qualité de service et le routage spécifique à VANET n'est pas explicite. Notre objectif secondaire est alors de promouvoir l'équité tout en supportant la QoS avec un algorithme de routage dynamique spécialisé pour VANET. L'équité et la priorité des classes de services de la QoS pour les différents types de flux seront mesurées en comparant les délais moyens pour les flux selon leurs différentes classes de services. Notons que les flux obtiennent des délais selon leur priorité et que les flux de même priorité obtiennent des délais selon leur utilisation de la bande passante relativement à leur juste part. De plus, des lacunes du protocole de routage dynamique par transmissions conditionnelles seront corrigées, c'est-à-dire que les données de trajectoires seront allégées par rapport à l'algorithme de référence des transmissions conditionnelles, et les hypothèses de trajectoires ne seront pas directement reliées à la vitesse des véhicules, comme c'est fait dans cet algorithme. En effet, l'algorithme de référence tient pour acquis que les véhicules arrêtés n'ont pas de trajectoire significative, alors qu'il est clair

pour nous qu'un véhicule à vitesse nulle peut quand même avoir un trajet et une destination définis et qu'il est tout simplement dans un bouchon de circulation.

4.3 Objectif secondaire : optimisation du routage dynamique avec équité et QoS

Cet objectif consiste à voir si on peut considérer l'équité et la QoS intégrées au routage dynamique de façon optimale. Les méthodes utilisées pour promouvoir l'équité et la qualité de service utilisent une partie de la bande passante disponible. Il est alors important de s'assurer que cette utilisation ne se fait pas au détriment des flux de communication et que la performance du système n'est pas affectée. Nous procédons par une comparaison entre l'utilisation de la bande passante avec les algorithmes de l'état de l'art et de déterminer si on doit faire des compromis pour utiliser la bande passante sans gaspillage.

Pour atteindre ces objectifs, nous apportons des changements aux algorithmes des techniques existantes dans notre revue de l'état de l'art, soit l'EDCF, la gestion des files d'attente TSFQ et les transmissions conditionnelles. Nous soulignons aussi les améliorations apportées par une architecture de communications intercouche qui s'applique à nos algorithmes parce que l'information qui détermine l'appartenance à un flux se situe en couche 4 alors que la technique EDCF qu'on propose de modifier se situe en couche 2. Pour valider notre nouvelle approche définie par un algorithme qui fait la promotion de l'équité des flux comme solution de gestion de la congestion dans les réseaux VANET, nous utilisons la technique de simulation et nous considérons les avantages sur quelques scénarios réalistes. Plus particulièrement nous considérons scénario de base étant décrit dans le chapitre de la problématique, soit la situation où les véhicules s'avertissent entre eux d'un bouchon de circulation afin d'offrir la possibilité aux véhicules approchant ce bouchon de prendre une trajectoire alternative.

Il faut constater qu'avec les situations de congestion, la communication sans fil se trouve dégradée [8]. Si l'on imagine la situation du bouchon expliquée dans le chapitre de la problématique, avec des groupements ou topologies temporaires de réseaux véhiculaires

s'échangeant des données de terrain, l'équité pourrait donner la chance aux communications importantes, mais non prioritaires, d'être captés par les véhicules, et ainsi leur permettre d'éviter le bouchon de justesse. En ce qui concerne notre scénario à simuler, il faut considérer un nombre important de véhicules dans le bouchon, utilisant la bande passante au maximum, et les autres véhicules qui arrivent au bouchon et qui peuvent servir de relais à ceux qui ont encore la chance de l'éviter. Avec l'équité des communications pour les flux qui utilisent peu de bande passante, les chances d'éviter le bouchon pour ces véhicules devraient augmenter, et c'est ce que nous comptons considérer dans ces travaux, pour valider l'efficacité de la QoS et de l'équité et leur performance.

Nous proposons d'intégrer nos solutions qui sont des modifications des algorithmes décrits dans l'état de l'art qui précède, intégrés dans un modèle intercouche afin d'optimiser les transmissions d'informations dans un scénario de congestion de circulation et de congestion du réseau de communications VANET. L'algorithme hybride d'ordonnancement des paquets en file d'attente TSFQ doit être intégré au mécanisme de l'EDCF pour que ce soit toujours le paquet provenant du flux le plus méritant qui soit transmis. Cette intégration nous permet de définir notre nouvel algorithme FDCF (*Fair Distributed Coordination Function*) qui sera présenté en détail dans la section suivante. Le surplus d'information du routage avec transmissions conditionnelles doit aussi transporter de l'information sur le retard des files d'attente du transmetteur précédent, afin de gérer la priorité de transmission non seulement d'une façon locale, mais aussi d'une façon qui implique la congestion des pairs sur le réseau. Par exemple si un nœud reçoit l'information sur le retard d'un pair lui indiquant qu'un flux est en attente de transmission sur ce véhicule, que ce flux utilise moins que sa juste part de bande passante et qu'il attend depuis plus longtemps que les flux en attente localement, le nœud local pourrait laisser la bande passante au nœud avec le flux injustement en attente. Les techniques d'intégration intercouches permettront d'exécuter ce plan, parce que la technique d'EDCF qui est utilisée pour gérer les délais avec des paramètres de délais au niveau de la couche d'accès et de contrôle de média sera modifiée pour permettre l'équité et la QoS (priorité) au niveau des

flux qui se situent au niveau la couche transport. Sans ces modifications, la gestion de la bande passante demeure au niveau des véhicules plutôt qu’au niveau des flux.

Nous utilisons le simulateur NS3 parce que c’est une nouvelle version du système NS2 qui est souvent cité dans la littérature comme on a constaté dans l’état de l’art. Le simulateur NS2 voit de moins en moins d’activités de recherche alors que l’utilisation du simulateur NS3 est en croissance [37]. Avec le simulateur NS3, on pourra assurément construire le scénario voulu. Il suffit de démontrer la différence dans les délais de réception avec le système de simulation sans nos algorithmes, et les délais de réception avec le système de simulation auquel on intègre nos algorithmes. De plus on pourra faire des statistiques démontrant qu’une plus grande proportion de véhicules aura la chance de prendre la voie de contournement.

4.3.1 Architecture de gestion équitable des flux sur VANET

Cette nouvelle architecture intercouche intègre les trois algorithmes retenus pour notre contribution proposée, soit l’algorithme FDCF, l’algorithme hybride de gestion des files d’attente pigeonier stratifié, et l’algorithme de routage des transmissions conditionnelles à trajet déterminé avec information de gestion d’équité et de QoS.

4.3.2 Algorithme de la FDCF

L’algorithme de la FDCF est à peu près le même que l’algorithme de la EDCF, sauf qu’on crée une nouvelle table pour permettre plus de classes de services et y ajouter une emphase sur la dimension équité en permettant d’ajouter un délai supplémentaire pour les flux qui utilisent plus que leur juste part de bande passante, et diminuer le délai pour les flux qui utilisent moins que leur juste part de bande passante. La fonction de transmission qui utilise et intègre l’algorithme de la FDCF ne sont pas décrites dans tous les détails afin d’alléger le texte, seul le pseudo-code pertinent se retrouve ici.

```
// tableau des AIFS (Adjusted Inter Frame Space  
// (espaces entre trames ajustés)  
// en unite de temps d’octets
```

```

const int AIFS [][] = { 2, 3, 6, 9 }
                      { 2, 4, 7, 9 }
                      { 2, 5, 8, 9 }

// tableau des fenetres de contention minimales
// (Contention Window) en unite de temps d'octets

const int CW_min [][] = { 3, 7, 15, 31 }
                        { 5, 9, 23, 39 }
                        { 7, 11, 31, 47 }

// tableau des fenetres de contention maximales
// (Contention Window) en unite de temps d'octets

const int CW_max [][] = { 7, 15, 1007, 1007 }
                        { 9, 19, 1015, 1015 }
                        { 11, 23, 1023, 1023 }

// init_retrait: fonction d'initialization des retraits
// parametres:
// classe: classe de service
// tflux : type de flux
// resultat: valeur initiale du retrait pour une classe donnee

int init_retrait ( int classe, int tflux )
{
    return ( CW_min [ classe ][ tflux ] +
            AIFS [ classe ][ tflux ] )
}

// augmenter_retrait: fonction d'augmentation des retraits
// parametres:
// classe: classe de service
// tflux : type de flux
// retrait: retrait en unite de temps d'octets
// resultat: valeur augmentee du retrait pour une classe donnee

int augmenter_retrait ( int classe, int tflux,
                      int retrait)
{
    retrait = retrait - AIFS [ classe ][ tflux ]
}

```

```

    retrait = retrait * 2
    retrait = retrait + 1

    if ( retrait > CW_max [ classe ][ tflux ] )
    {
        retrait = CW_max [ classe ][ tflux ]
    }

    return ( retrait + AIFS [ classe ][ tflux ] )
}

// transmission: fonction de transmission de paquets décrite
// minimalement ici avec l'utilisation d'une structure de
// donnees abstraites paquet qui contient un champ retrait de
// type int et une variable collision de type int aussi
// la fonction de transmission n'est pas decrite completement
// afin d'alléger le texte

void transmission ( struct packet_t *packet )
{
    int collision = FAUX; // variable booleenne
    int transmit = FAUX; // variable booleenne

    ...

    paquet->retrait = initialiser_retrait ( classe, tflux )

    ...

    while ( !transmit )
    {

        attendre ( paquet->retrait )

        collision = verifier_collision( );

        if ( collision )

            paquet->retrait =
                augmenter_retrait ( classe, tflux,
                                    paquet->retrait )

        else

```



```

        paquet->retrait = initialiser_retrait ( classe,
                                                tflux )
    ...
    transmettre ( paquet )
    ...
}

```

Figure 12 algorithme FDCF

4.3.3 Algorithme hybride de gestion des files d'attente pigeonier stratifié

L'algorithme hybride de gestion des files d'attente est un algorithme d'ordonnement basé sur le tri pigeonier (bin sort) et l'algorithme d'ordonnement stratifié avec simplification des paramètres. Le tri de pigeonier permet le classement des flux qui se fait en temps constant pour chaque élément (temps constant par élément ou $O(n)$ pour le vecteur) parce que les messages sont classés dans la bonne case pour leur flux dès leur présence ou le changement de comportement du flux, alors que l'algorithme avec tri s'exécute en temps logarithmique ($O(\log n)$ par élément ou $O(n \log n)$ pour le vecteur). La solution avec le tri de pigeonier est donc moins générale parce qu'elle dépend de la possibilité de mettre les éléments dans des cases et les frontières de cases doivent aussi être connues, mais elle est plus efficace quand on peut le faire. En paramétrant l'échelle utilisée pour déterminer les frontières des cases du pigeonier, on peut approcher la flexibilité du tri général.

```

// pigeonner_flux: fonction qui determine dans quelle case du
// pigeonier mettre le flux en parametre
// parametres:
// flux: flux dont on veut determiner la case
// resultat: index de case du pigeonier

int pigeonner_flux ( struct flux_t *flux )
{
    int quant

    quant = simplifier_temps ( flux->paquet_tete.arrivee )

```

```

        + simplifier_taille ( flux->paquet_tete.taille )
        + simplifier_taille ( flux->taille_totale )
        + flux_A->classe

    return quant / echelle_globale
}

// traitement: fonction de traitement de paquets décrite
// minimalement ici avec l'utilisation d'une structure de
// donnee abstraite paquet, une structure de donnee abstraite
// flux qui est une liste de paquets, ainsi qu'un vecteur de
// flux la fonction de transmission n'est pas decrite
// completement afin d'allegger le texte

void traitement ( void )
{
    ...

    packet = reception()

    ...

    if ( packet )
    {
        ...

        flux = introduire_paquet ( packet )

        ...

        pigeonner_flux ( vecteur_flux, flux )

        ...
    }
    ...

    packet = extraire_packet ( vecteur_flux )

    ...

    if ( packet )

```

```

    {
        ...
        transmission ( packet )
        ...
    }
}

```

Figure 13 algorithme BTSFQ

4.3.4 Algorithme des transmissions conditionnelles à trajet déterminé

Cet algorithme est celui des transmissions conditionnelles avec simplification du fait qu'il n'a besoin que du vecteur de trajectoire du véhicule, et non des 10 dernières lectures de localisation par GPS. De plus, cet algorithme considère que les véhicules arrêtés ont quand même une trajectoire valide.

```

// verifier_condition: fonction qui verifie les conditions de
// transmission d'un paquet selon le protocole de routage
// dynamique des transmissions conditionnelles
// parametres:
// packet: pointeur sur le paquet a verifier
// resultat: entier representant une valeur booleene
// a savoir si la condition de transmission est remplie

int verifier_condition ( struct packet_s *packet )
{
    switch ( packet->condition.type )
    {
        case MEME_TRAJET :

            if ( comparer_trajet ( paquet->condition.trajet,
                                vehicule.trajet ) )
                return ( FAUX )

            if ( comparer_distance ( paquet->condition.distance,
                                vehicule.position)

```

```

        return ( FAUX )

    return ( VRAI )

    case TRAJET_DIFFERENT :

        ...

    }
}

void transmission ( void )
{

    ...

    transmettable = verifier_condition ( packet )

    ...

    if ( transmettable )

        transmettre ( packet )

    else

        rejeter ( packet )

    ...

}

```

Figure 14 algorithme des transmissions conditionnelles à trajet déterminées

4.4 Diagramme du modèles des couches

Le diagramme qui suit présente en synthèse le modèle de couche et interactions inter-couche que nous présentons dans nos travaux.

Couche	Technologies Rejetés	Technologies État de l'art	Technologies Mémoire	Intercouche
1- Physique	CDMA / IDMA	DSRC		
2- Accès		EDCF / TSFQ	FDCF / BSTFQ	Couche 4
3- Réseau		IP		
4- Routage	Convois Dynamiques	TCP	Transmissions Conditionnelles	Couche 2
5- Application			Contournement	

Figure 15 diagramme de couches

4.5 Présentation du système de simulation

Pour simuler à la fois le réseau de communication basé sur la norme VANET et son implémentation dans un modèle réaliste d'autoroute avec congestion de véhicule, il est important de définir un cadre de travail adéquat pour notre étude. Ce cadre doit permettre de faire de la simulation de trafic sur un tronçon autoroutier avec un modèle de communication v&v complet et réaliste. Nous avons identifié le cadre de travail proposé dans [37], car il répond à ces deux besoins, la mobilité des véhicules et la communication entre eux. Celui-ci est offert sur Internet et son modèle inclut une section d'autoroute avec une configuration des voies et différents types de véhicule présentant des comportements différents, ainsi qu'une interaction de ceux-ci avec les messages provenant du réseau VANET. Ce modèle, basé sur le modèle véhiculaire IDM (*Intelligent Driver Model*) proposé dans [38], aura cependant besoin de quelques modifications pour représenter notre scénario de sortie d'autoroute. Ce modèle tel que fourni par les chercheurs qui développent le cadre de travail présente une section d'autoroute à un nombre variable de voies, dans deux sens, alors que nous avons besoin d'un seul sens et d'une voie de sortie. De plus, le modèle et son cadre de travail sont implémentés sur la plateforme de simulation de réseaux ns-3 [39].

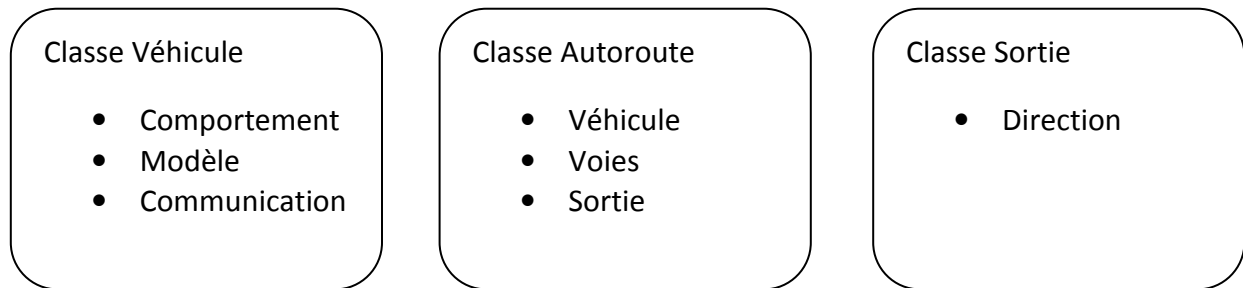


Figure 16 extension du modèle de base

Dans la Figure 16 extension du modèle de base on voit l'addition du champ sortie pour la classe autoroute. Cette extension au modèle est similaire à celle du modèle IDM de base, pour lequel la classe entrée existe [38].

Le système de simulation peut être téléchargé à partir de la page web [40]. Nous utilisons la version Linux Fedora pour compiler et exécuter le simulateur. Nous y avons intégré le modèle de base que nous avons téléchargé à partir de la page web [41]. Le code a dû être modifié quelque peu parce que le compilateur c++ que nous utilisons est plus récent que celui utilisé par les créateurs du code original, nous utilisons la version 4-7-2. Les modifications sont principalement des sélections explicites de hiérarchie d'objet parce que le compilateur se plaint de ne pas reconnaître les méthodes invoquées, alors qu'elles sont des méthodes standard. Le modèle a été ajouté au répertoire d'exemple, faute de savoir comment créer un nouveau répertoire. Le modèle roule en utilisant la commande :

```

./waf --run `vanet-highway-test --project=examples/vanet-
highway-version2/Examples/SimpleStraightHighway.xml --
enablevehiclereceive=1`
  
```

Cette commande invoque le simulateur ns-3 avec le modèle que nous avons téléchargé. Le modèle produit deux fichiers traces qui peuvent être analysés par la suite. Ces fichiers sont networkTrace.csv qui relatent les messages sur le réseau sans fil et vehiculeTrace.csv qui relatent les positionnements des véhicules. C'est l'analyse de ces fichiers qui nous permettra

de vérifier que le modèle fonctionne bien, et valider que la nouvelle architecture proposée avec ses algorithmes modifiés fasse une différence au niveau des délais pour l'équité et la qualité de service en rapport à nos objectifs. De plus on pourra constater l'amélioration des conditions de bouchon pour une bonne partie des automobilistes.

4.6 Implémentation du scénario

L'implémentation du scénario complète le modèle choisi pour y ajouter la sortie d'autoroute et modifie les communications avec l'implémentation des algorithmes qui font la promotion d'une meilleure équité et QoS de façon optimale. Dans le scénario qui suit en Figure 17 sortie de route, les véhicules qui entrent sur le tronçon en bas à gauche sont dans la possibilité de prendre une décision, soit continuer sur la route ou prendre la sortie. Ils ne peuvent voir le bouchon de circulation qui s'est formé parce, mais la communication v&v peut leur permettre d'en avoir connaissance et prendre une décision de contournement.

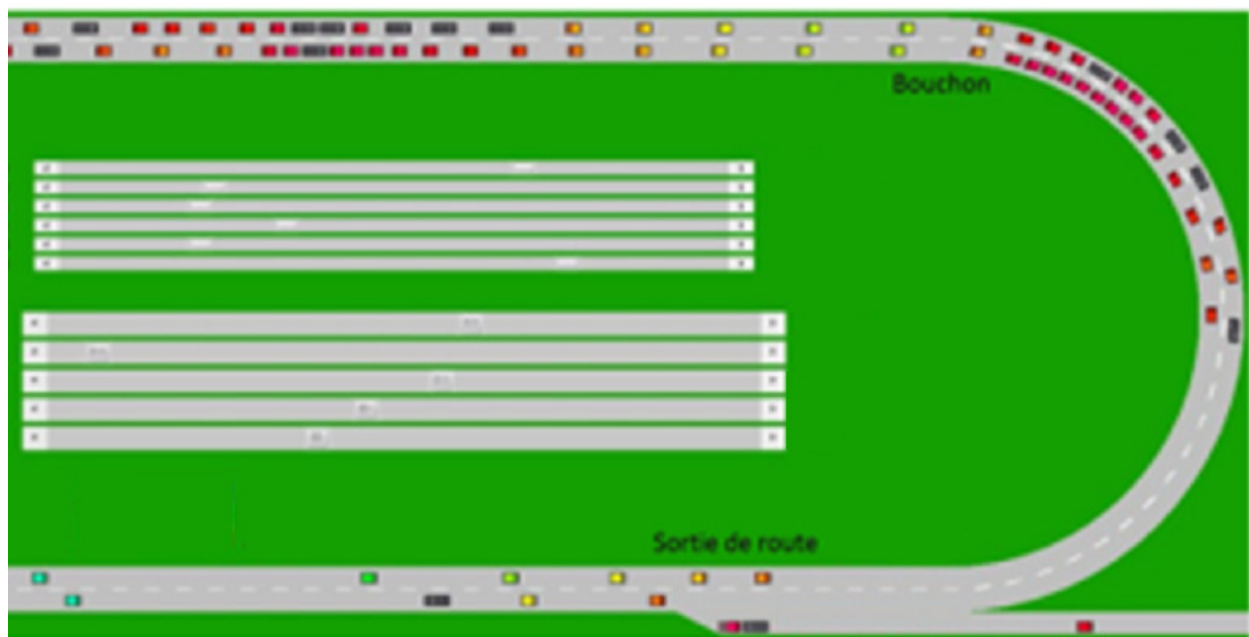


Figure 17 sortie de route

4.7 Expérimentation

Pour valider notre approche et souligner notre contribution, nous établiront des tests pour comparer les délais de transmission obtenus sur le système de simulation avec notre scénario implanté, pour un système qui utilise les algorithmes existants et ceux que nous avons définis pour notre approche.

4.7.1 Mesures sur l'équité des flux

Les délais équitables pour les flux de communication se mesureront comme suit. Les flux de communication qui utilisent moins que leur juste part de bande passante auront de meilleurs délais, soit des délais plus courts que les flux de communication qui utilisent plus que leur juste part de bande passante. De plus il faudra démontrer que ces délais sont bornés.

4.7.2 Adéquation de l'équité dans le modèle de QoS

Nous proposons de vérifier que les flux de communication prioritaires ont de meilleurs délais que les flux de communication de type meilleur effort, alors que les flux de communication meilleur effort ont des délais équitables soit selon leur utilisation de la bande passante. Les délais seront calculés en comparant le timbre temporel d'envoi original des messages avec celui de réception.

4.7.3 Optimisation de l'équité et de la QoS

Nous proposons de vérifier que l'équité et la QoS sont rendues de façon optimale, c'est-à-dire d'une façon qui conserve le travail (*work conserving*). Il s'agit de démontrer que les délais d'utilisation de la bande passante ne sont jamais plus grands qu'avec les algorithmes de l'état de l'art. Encore une fois les timbres temporels d'envoi et de réception seront utilisés pour calculer les délais.

5 Réalisation

5.1 Configuration du système d'expérimentation, défis et solutions

Le cadre d'expérimentation basé sur le simulateur ns-3 et le modèle IDM (Intelligent Driver Model) que nous avons identifié dès le début du projet s'avère pertinent pour tester et valider nos solutions algorithmiques. Cependant, de nombreuses adaptations et configurations étaient requises pour garantir la fonctionnalité et la crédibilité des résultats expérimentaux. Le plan était de vérifier s'il avait problème avec le saut de version par rapport à la version de ns-3 utilisée par les créateurs du modèle IDM, et dans le cas où il n'y en aurait pas, de bénéficier des développements et possiblement du support de la communauté. Les problèmes rencontrés étaient dus non pas au fait que ns-3 ait évolué de plusieurs versions, mais plutôt que le compilateur c++ a changé depuis le temps. Les adaptations que nous avons réalisées sur IDM sont particulièrement reliées au compilateur c++.

Relativement à la configuration, le langage qui est basé sur XML n'est pas documenté pour le modèle IDM. Il a fallu dériver le fonctionnement de la syntaxe de configuration ou lire le code pour comprendre le comportement de la configuration.

On a pu faire fonctionner la configuration avec des petites distances, 1 kilomètre avant la sortie et 1 kilomètre après la sortie et relativement peu de véhicules et progressivement ajouter de la distance, du temps et des véhicules communiquant, pour passer à notre scénario avec 5 kilomètres avant la sortie, 5 kilomètres après la sortie, une sortie de contournement avec un angle de 45 degré, deux voies d'autoroute et un bouchon créé quand des véhicules spécifiques se rendent après le cinquième kilomètre après la sortie. Relativement à la durée de la simulation, 30 minutes de temps réel sont représentées. Cependant des modifications ont été requises au niveau des bibliothèques intégrées dans ns-3 telles que le modèle Yan's WiFi de ns-3 pour avoir plus de communication, plus de congestion et plus de temps parce que le modèle tel que fournit ne permet pas d'avoir des messages en file d'attente, le code génère une assertion

si des messages doivent occuper la file d'attente de la couche 2. Quelques assertions du code ont du être enlevées pour permettre l'acceptation de paquets en file d'attente parce que les assertions n'étaient plus valables pour les nouvelles implémentations des modèles. Le modèle IDM à été codé alors que les modèles de DSRC n'étaient pas encore disponibles, notamment les particularités des mises-a-jour pour le modèle de qualité de service n'était pas utilisées, donc le code a du être modifié pour être mis à jour pour qu'on puisse valider les effets des algorithmes sur les délais.

Nous avons considéré un outil de visualisation écrit en java permet de visionner la configuration et le déplacement des véhicules pendant la simulation. Cet outil permet de confirmer que la configuration de la simulation est bonne de façon visuelle. Cet outil utilise le fichier de trace des véhicules pour afficher le résultat de simulation de façon animée.

Le modèle IDM intégré au simulateur ns-3 permet de générer un fichier de sortie permettant de collecter les résultats des simulations qui permettent de valider les approches algorithmiques proposées dans ce projet. Contrairement aux objectifs des auteurs du modèle IDM qui se concentraient sur le comportement des véhicules intégré au simulateur de réseau, notre intérêt se situe au niveau des paquets, à l'identification des paquets envoyés et reçus ainsi que les temps respectifs de ces paquets pour chaque véhicule afin de calculer les délais moyens qui permettent les mesures de l'équité et de la qualité du service. Le format du fichier trace de simulation à été corrigé pour inclure le numéro de séquence ou de référence pour les paquets. Cette information est nécessaire pour faire le lien entre un paquet envoyé et reçu afin de calculer le délai. Les informations de position de chaque paquet ont aussi du être corrigées dans le fichier de trace. Il a fallu corriger ces deux problèmes majeurs de la génération du fichier de trace de réseau. Une fois ces problèmes corrigés, il fallait aussi créer un outil pour filtrer les informations de ce fichier afin de calculer les délais moyens de réception de chaque paquet ainsi que les écarts types, le nombre de réceptions fructueuses et infructueuses. Cet outil à été crée en utilisant le langage awk. L'infrastructure a été complétée avec le codage des

algorithmes de l'état de l'art. Notons aussi que l'infrastructure garantie des délais bornés pour les besoins de l'équité vu que les files d'attente sont limitées à 30 éléments et l'algorithme de la EDCF abandonne la transmission une fois la fenêtre de contention maximale est expirée.

5.2 Résultats d'expérimentation

Nous présentons dans cette section nos résultats d'expérimentation ainsi qu'une analyse de l'effet des algorithmes que nous proposons sur les délais de transmission. Le tableau suivant présente les délais de transmissions basés sur plusieurs critères pour un même scénario, de façon statistique selon les combinaisons de cas du même scénario. C'est-à-dire que le scénario avec sortie de route est toujours simulé mais avec des configurations différentes, et dans ces contextes les délais des différentes classes de services sont calculés. Les expérimentations sont ainsi regroupées en deux groupes, c'est-à-dire sans les transmissions conditionnelles activées et avec transmissions conditionnelles activées. Ces deux groupes sont divisés en deux sous groupes, c'est-à-dire sans QoS, avec QoS et avec QoS et équité. L'illustration qui suit ce tableau rappelle le scénario tel que décrit dans le chapitre précédent, cette fois en utilisant l'outil de visualisation qui vient avec l'infrastructure IDM.

	Classes de service	Types de flux
Sans transmissions conditionnelles sans qualité de service	VO (voix), VI (vidéo), BE (meilleur effort), BK (arrière-plan)	LE (moins de bande), FA (juste part), MO (plus de bande)
Sans transmissions conditionnelles avec qualité de service	VO (voix), VI (vidéo), BE (meilleur effort), BK (arrière-plan)	LE (moins de bande), FA (juste part), MO (plus de bande)
Sans transmissions conditionnelles avec qualité de service et équité	VO (voix), VI (vidéo), BE (meilleur effort), BK (arrière-plan)	LE (moins de bande), FA (juste part), MO (plus de bande)
Avec transmissions conditionnelles	VO (voix), VI (vidéo), BE (meilleur effort), BK (arrière-plan)	LE (moins de bande), FA (juste part), MO (plus de bande)
Avec transmissions conditionnelles avec qualité de service	VO (voix), VI (vidéo), BE (meilleur effort), BK (arrière-plan)	LE (moins de bande), FA (juste part), MO (plus de bande)
Avec transmissions conditionnelles avec qualité de service et équité	VO (voix), VI (vidéo), BE (meilleur effort), BK (arrière-plan)	LE (moins de bande), FA (juste part), MO (plus de bande)

Figure 18 tableau des options de scenario

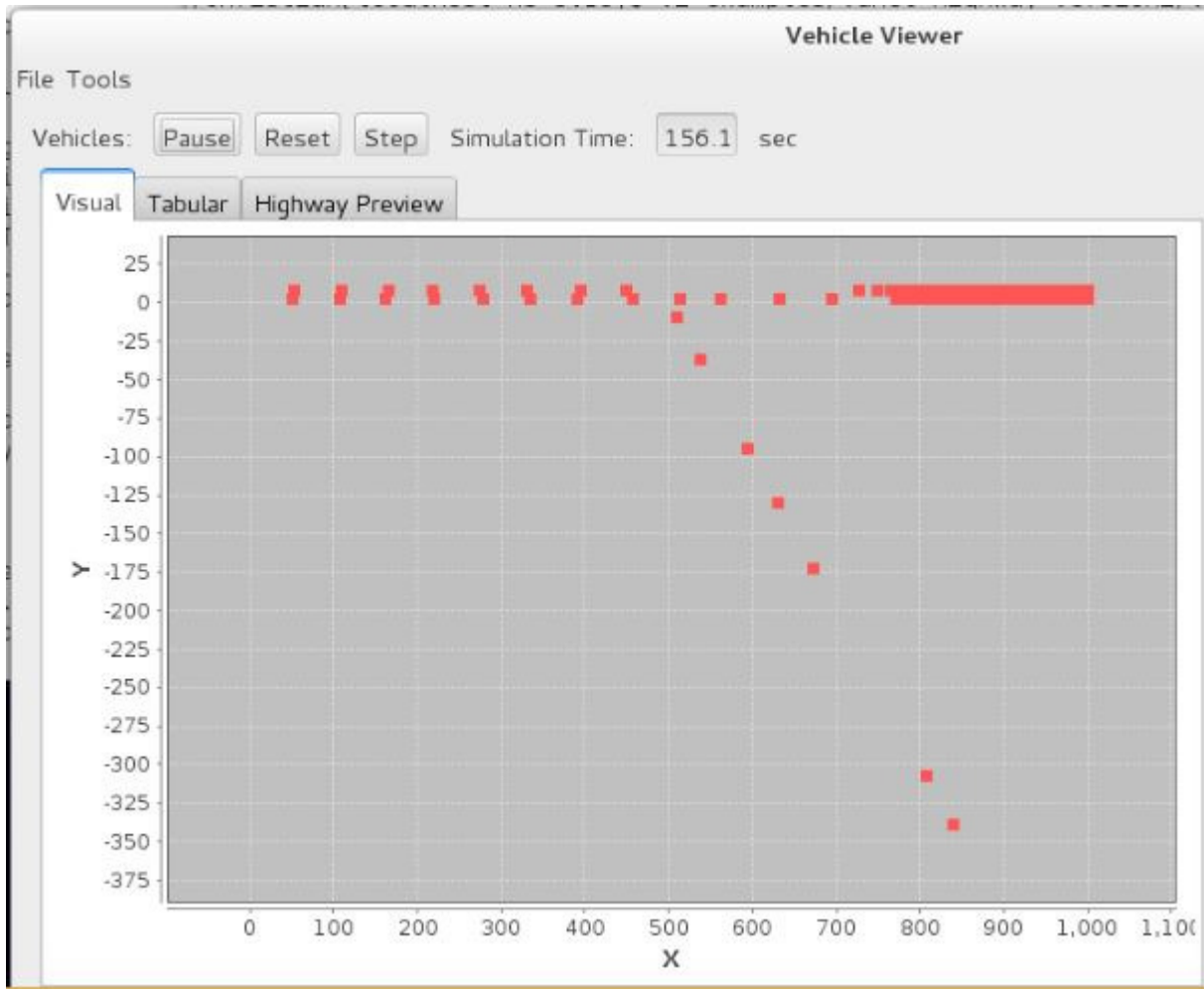


Figure 19 visualisation du scénario avec l'outil Vehicle Viewer

5.2.1 Résultats sans l'algorithme des transmissions conditionnelles

L'algorithme de routage par la méthode des transmissions conditionnelles n'est pas utilisé dans ces premiers résultats. C'est cet algorithme que nous décrivons dans le chapitre précédent qui permettra le routage des paquets de l'application de détection de bouchon. Nous constatons les délais sans cet algorithme et sans les transmissions conditionnelles dans cette section.

Sans l'algorithme des transmissions conditionnelles activé, nous exécutons la simulation pour constater les effets sur les délais des algorithmes de la FDCF et de la BTSFQ comme décrits dans le chapitre précédent. Par défaut le système de simulation vient sans la QoS activée, donc les

algorithmes utilisés sont la DCF et FIFO tels que décrits dans le chapitre précédent pour la discipline de gestion de file d'attente, donc nous exécutons la simulation sous ces paramètres pour constater les délais. Par la suite nous activons la QoS avec la EDCF et TSFQ tels que décrits dans le chapitre précédent qui sont une option du système ns-3 pour constater les délais à nouveau. Finalement, nous activons nos algorithmes, la FDCF et la BTSFQ pour constater les délais améliorés par ces algorithmes. Nous présentons deux séries de tableau pour ces résultats, la première série montre ces résultats groupés par classes de service, et la deuxième série montre ces résultats groupés par type de flux.

Dans notre scénario avec la sortie de route, une demi-heure de circulation routière est simulée, et le temps pris pour exécuter cette simulation est environ douze heures pour chaque jeu d'algorithmes.

5.2.1.1 Délais par classe de service sans QoS

Classe de service	Paquets Arrivés	Paquets Perdus	Délais moyen ns
VO (voix)	2634058	2816427	2726053
VI (vidéo)	2583459	2751974	2751253
BE (meilleur effort)	2621154	2795260	2746825
BK (arrière-plan)	2640306	2808121	2718215

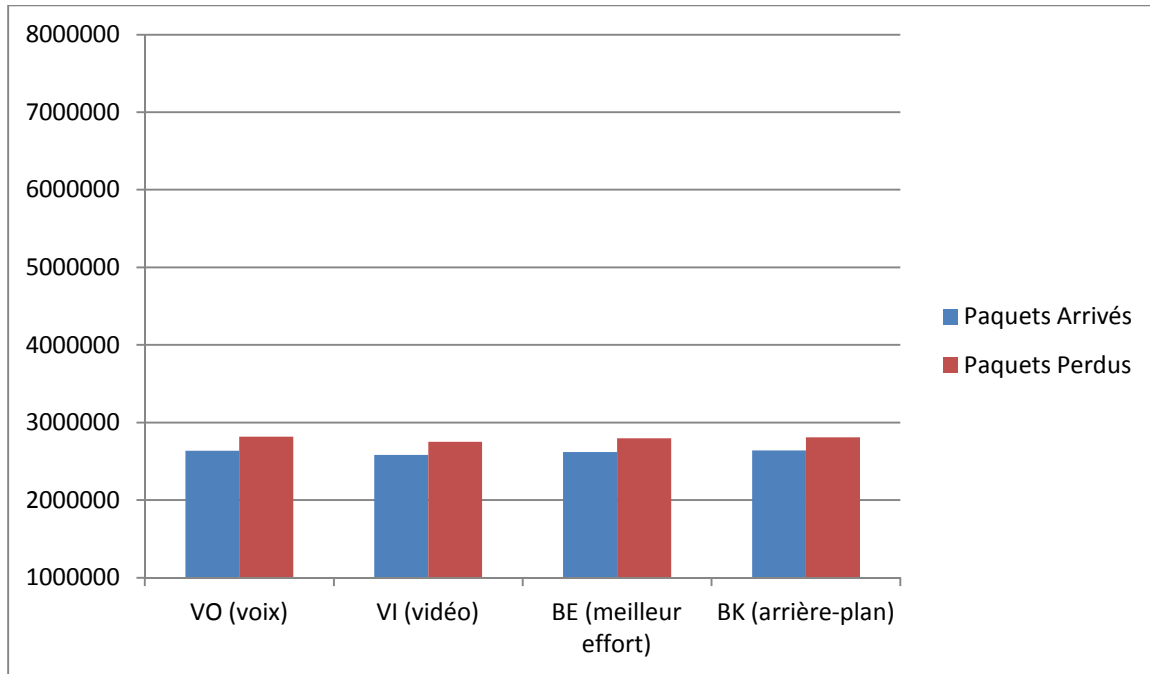


Figure 20 paquets par classe de service sans QoS sans transmissions conditionnelles

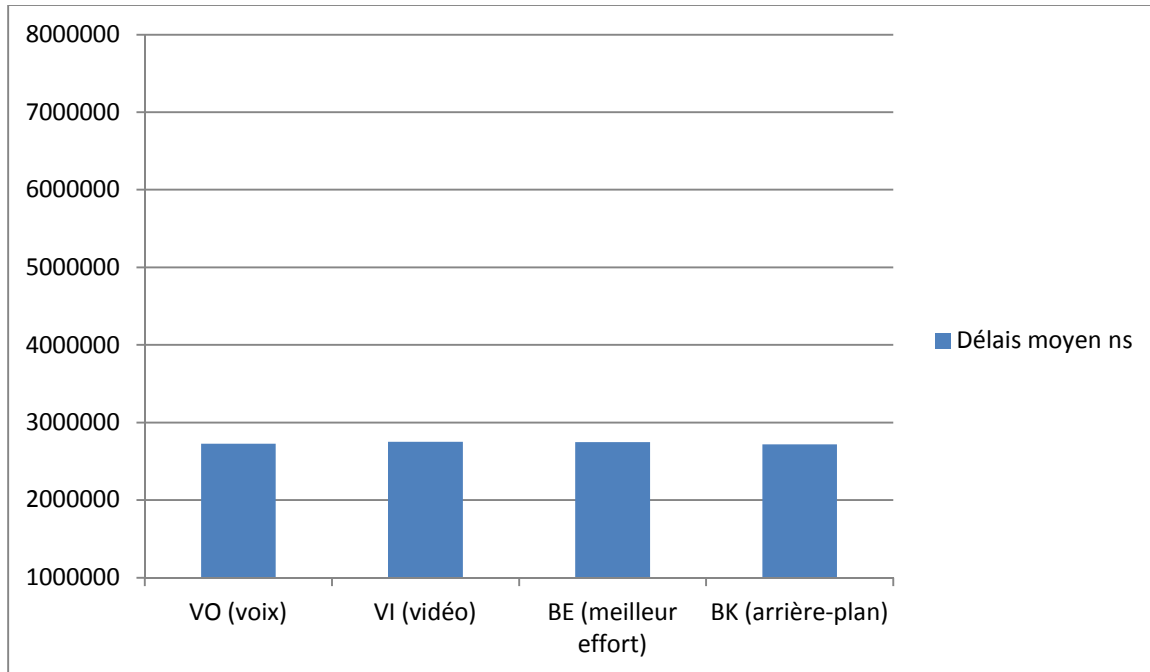


Figure 21 délais par classe de service sans QoS sans transmissions conditionnelles

5.2.1.2 Délais par classe de service avec QoS

Classe	Paquets Arrivés	Paquets Perdus	Délais moyen ns
VO (voix)	2625425	2833940	1697020
VI (vidéo)	2575292	2751155	2478381
BE (meilleur effort)	2616845	2784985	2884837
BK (arrière-plan)	2642001	2805098	3497462

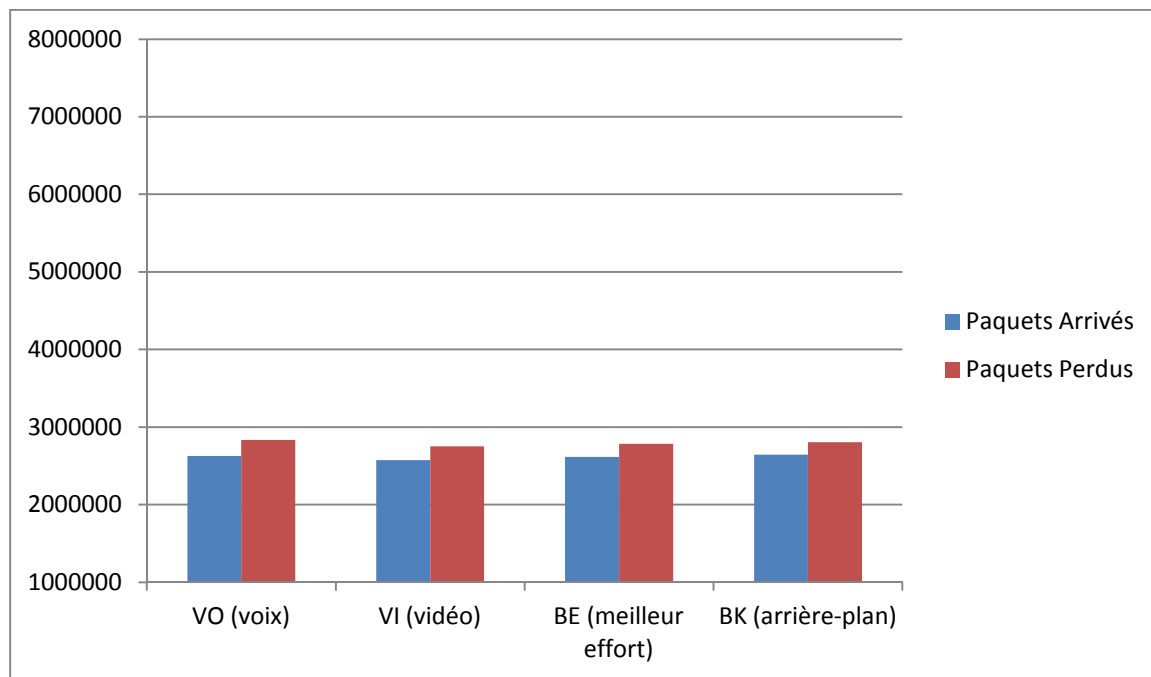


Figure 22 paquets par classe de service avec QoS sans transmissions conditionnelles

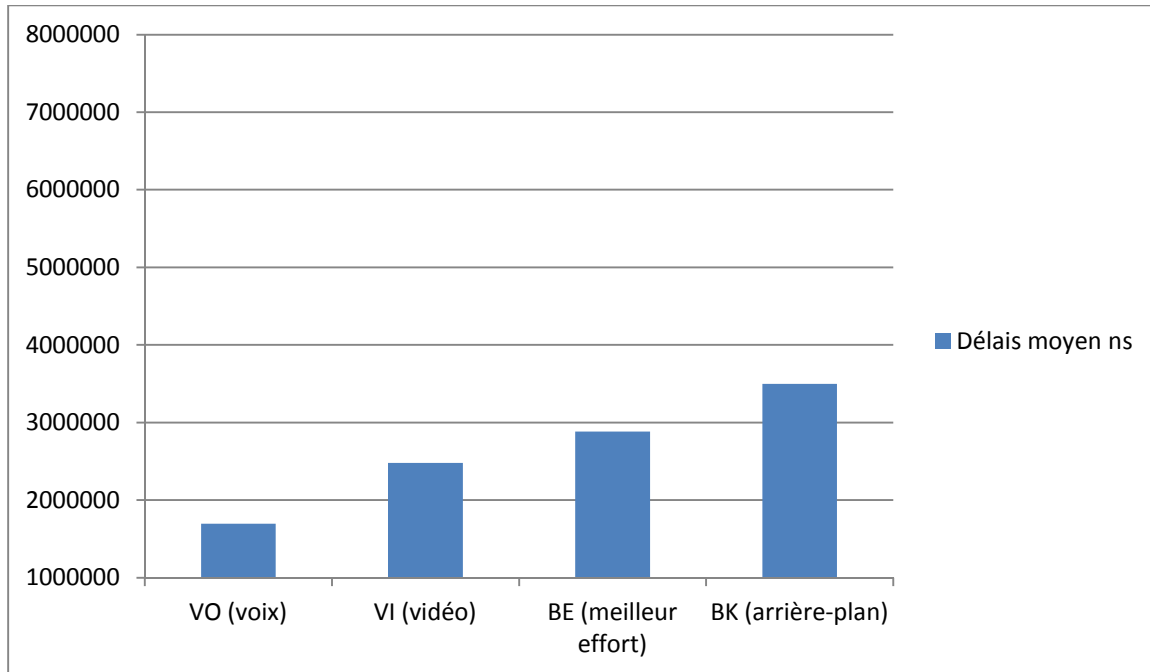


Figure 23 délais par classe de service avec QoS sans transmissions conditionnelles

On remarque que la QoS améliore les délais pour les classes prioritaires à gauche du graphique, respectivement VO et VI et que les délais sont croissants en allant vers la droite du graphique pour les classes BE et BK. Quand la QoS n'était pas activée avec le graphique précédent, on voyait que les délais étaient plus ou moins égaux entre les classes de service.

5.2.1.3 Délais par classe de service avec QoS et Équité

Classe	Paquets Arrivés	Paquets Perdus	Délais moyen ns
VO (voix)	2271432	2489529	1683406
VI (vidéo)	2241624	2450723	2535020
BE (meilleur effort)	2266166	2459159	2947883
BK (arrière-plan)	2269007	2465306	3671616

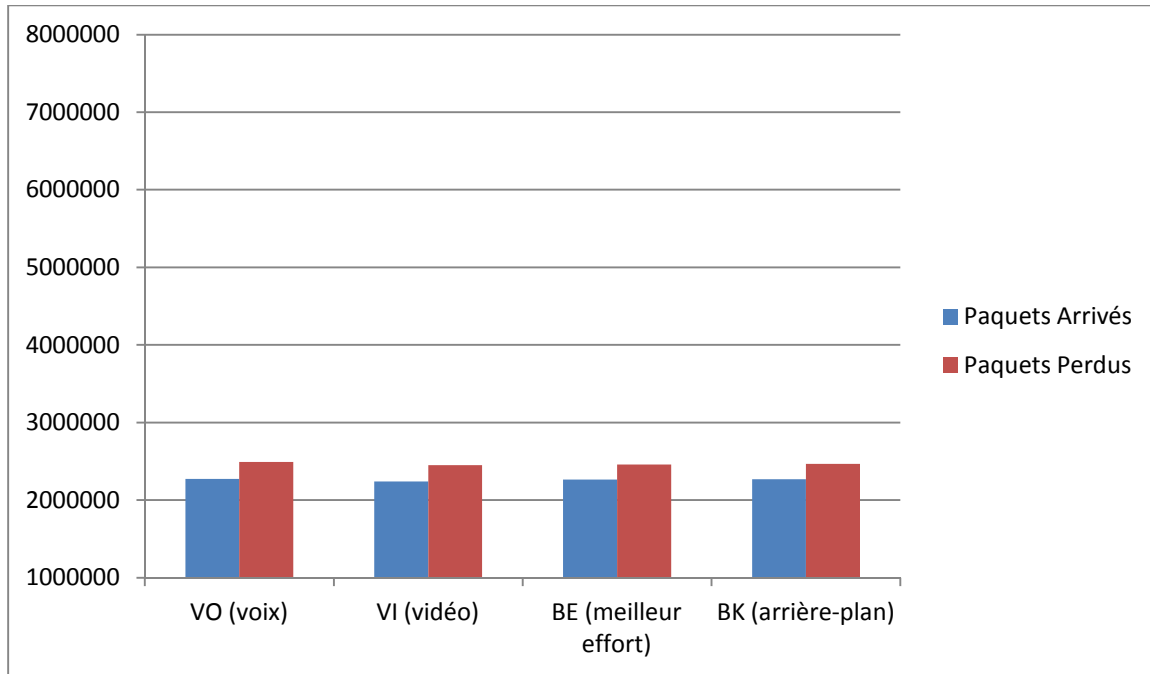


Figure 24 paquets par classe de service avec QoS et équité sans transmissions conditionnelles

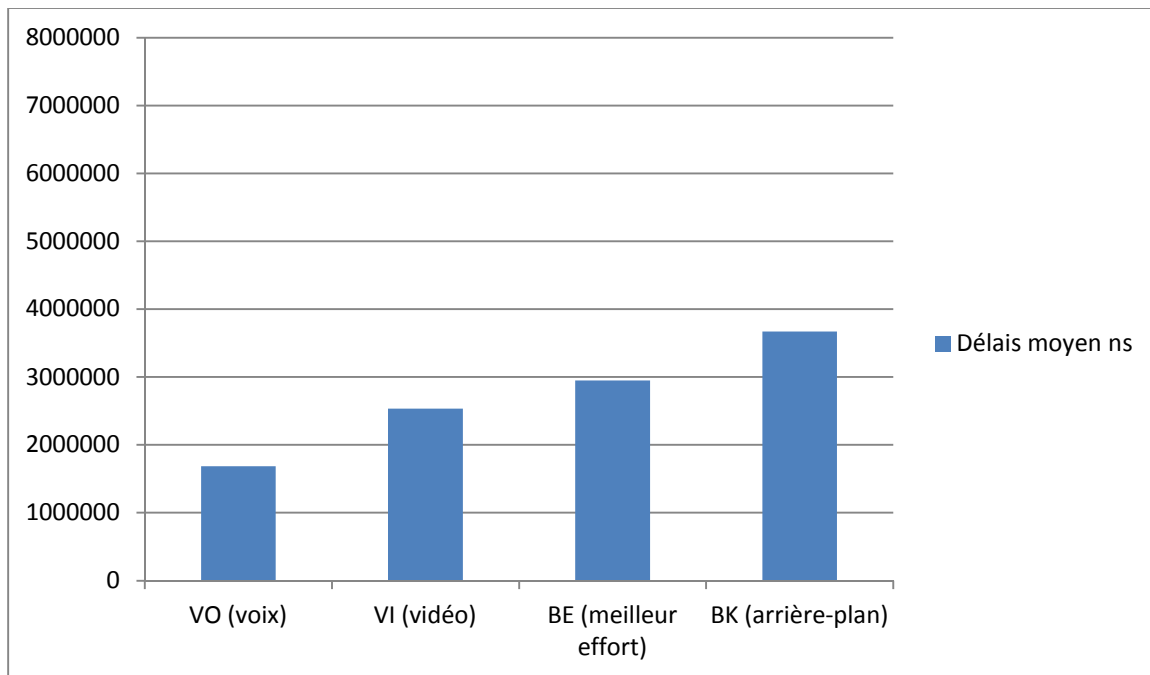


Figure 25 délais par classe de service avec QoS et équité sans transmissions conditionnelles

On remarque qu'en général la moitié des paquets est perdue, ce qui démontre bien une situation de congestion sur le réseau de données. On remarque aussi que les délais croissent avec les priorités de classes de service qui démontre que la QoS fonctionne bien dans tous les cas et que l'équité fournie par les algorithmes FDCF et BSTFQ n'affecte pas la QoS négativement.

5.2.1.4 Délais par type de flux sans QoS

Type	Paquets Arrivés	Paquets Perdus	Délais moyen ns
LE (peu de bande)	1505189	1619147	4751256
FA (juste part)	2327541	2499086	3511734
MO (+ de bande)	6646247	7053549	2007127

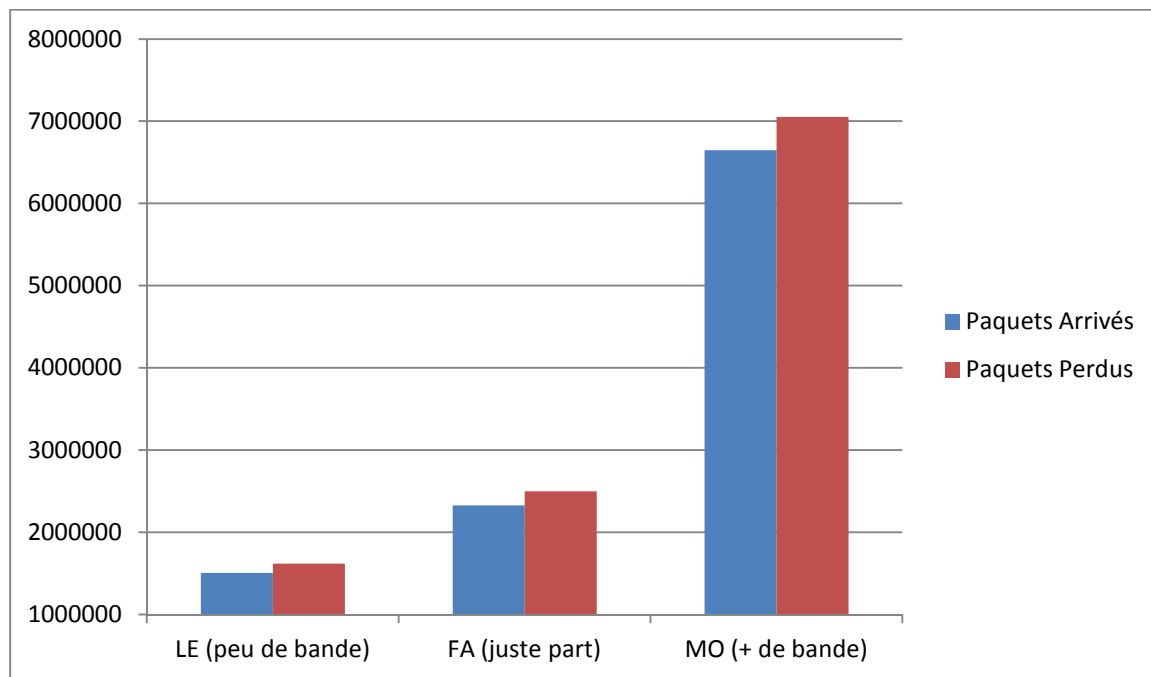


Figure 26 paquets par types de flux sans QoS sans transmissions conditionnelles

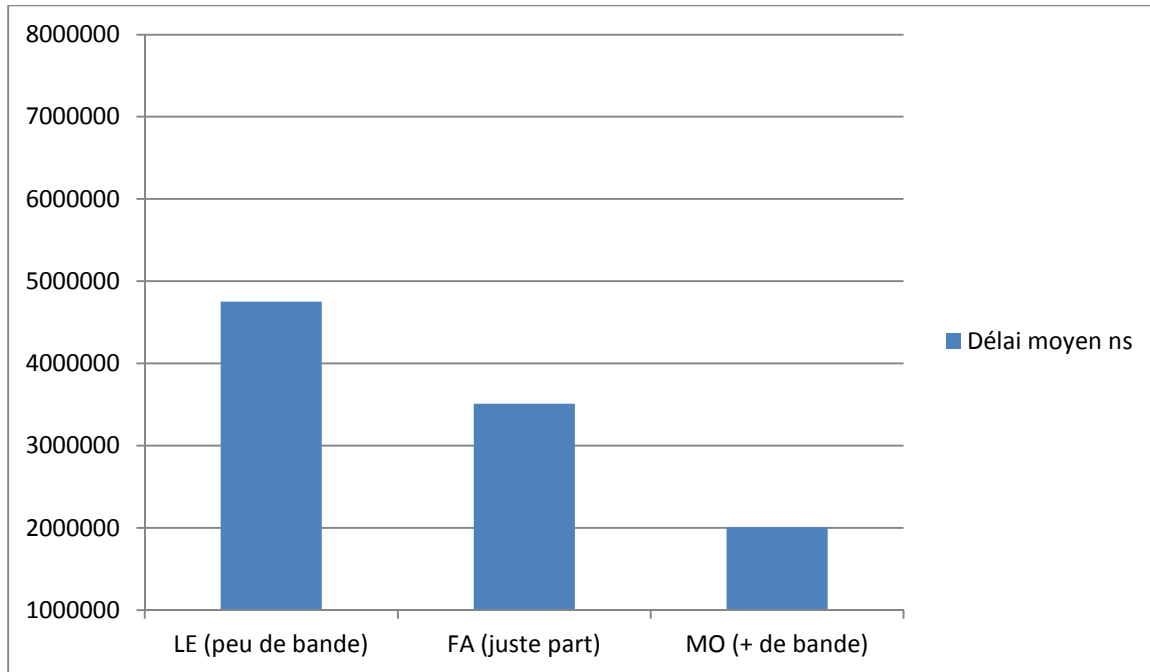


Figure 27 délais par type de flux sans QoS sans transmissions conditionnelles

On remarque ici que les délais sont inversés, comme ni la QoS ou l'équité sont activés, c'est-à-dire que les flux qui utilisent beaucoup de bande passante, plus que leur juste part, sont avantagés, et les flux qui utilisent peu de bande passante sont pénalisés. On remarque aussi que la quantité des paquets arrivés et des paquets perdus est proportionnel au type de flux par rapport à l'utilisation de la bande passante.

5.2.1.5 Délais par type de flux avec QoS

Type	Paquets Arrivés	Paquets Perdus	Délais moyen ns
LE (peu de bande)	1495352	1607302	3075512
FA (juste part)	2319822	2490145	2783465
MO (+ de bande)	6644389	7077731	2494030

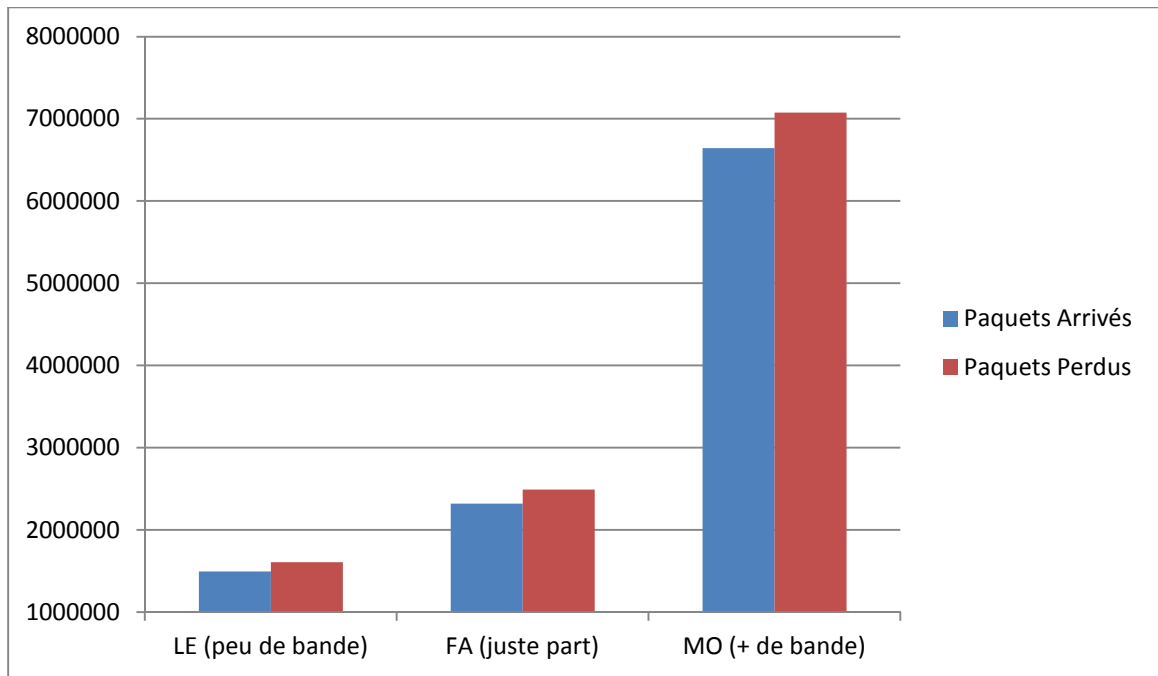


Figure 28 paquets par types de flux avec QoS sans transmissions conditionnelles

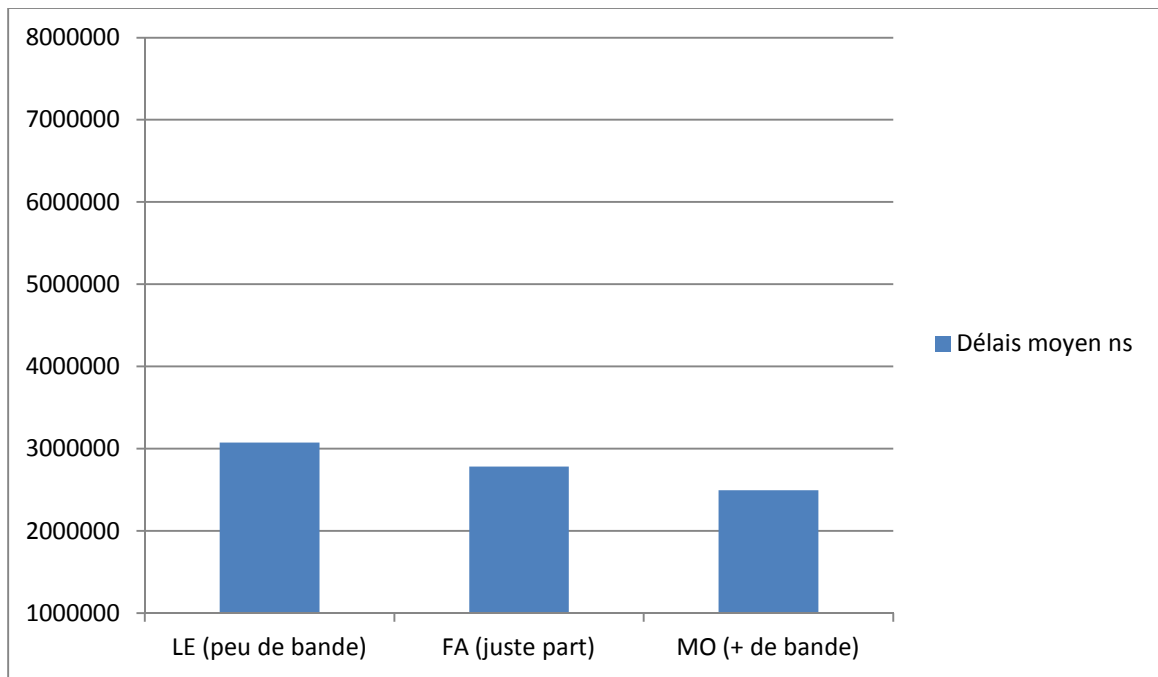


Figure 29 délais par type de flux avec QoS sans transmissions conditionnelles

La QoS activée ici stabilise les délais mais ils sont toujours inversés par rapport au concept d'équité fournit par les algorithmes FDCF et BSTFQ. Les délais des flux utilisant peu de bande passante sont désavantagés par rapport à ceux utilisant plus que leur juste part de bande passante.

5.2.1.6 Délais par type de flux avec QoS et Équité

Type	Paquets Arrivés	Paquets Perdus	Délais moyen ns
LE (peu de bande)	1287476	1411266	2089356
FA (juste part)	2009707	2200328	2412446
MO (+ de bande)	5751046	6253123	2952386

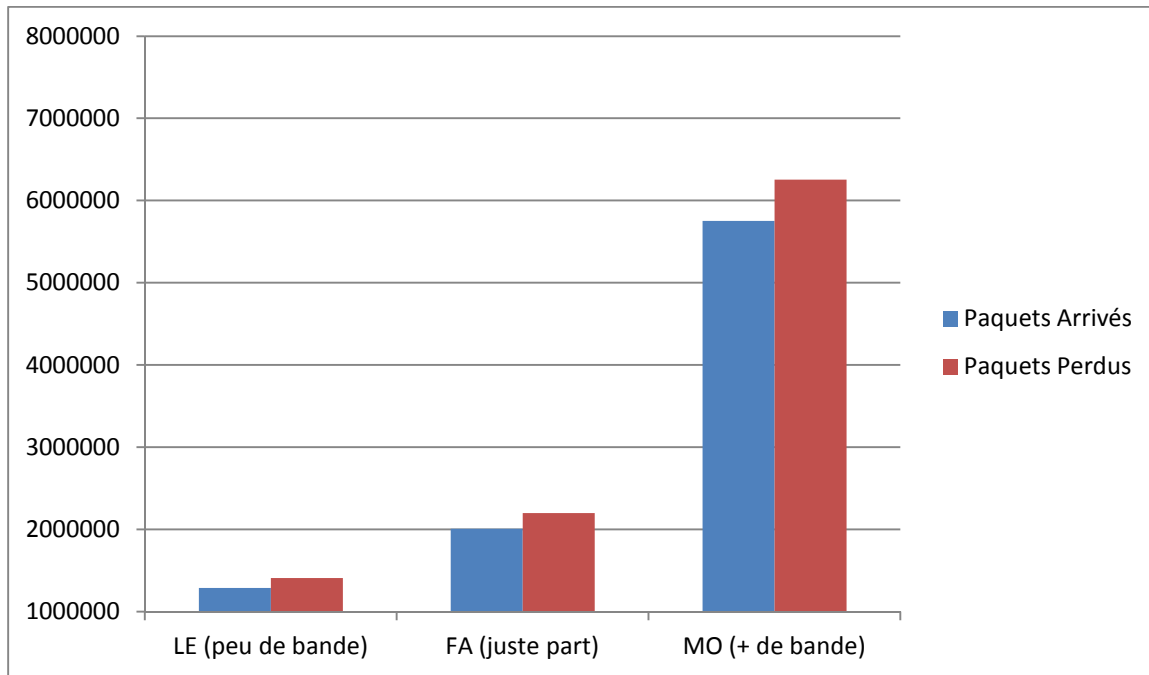


Figure 30 paquets par types de flux avec QoS et équité sans transmissions conditionnelles

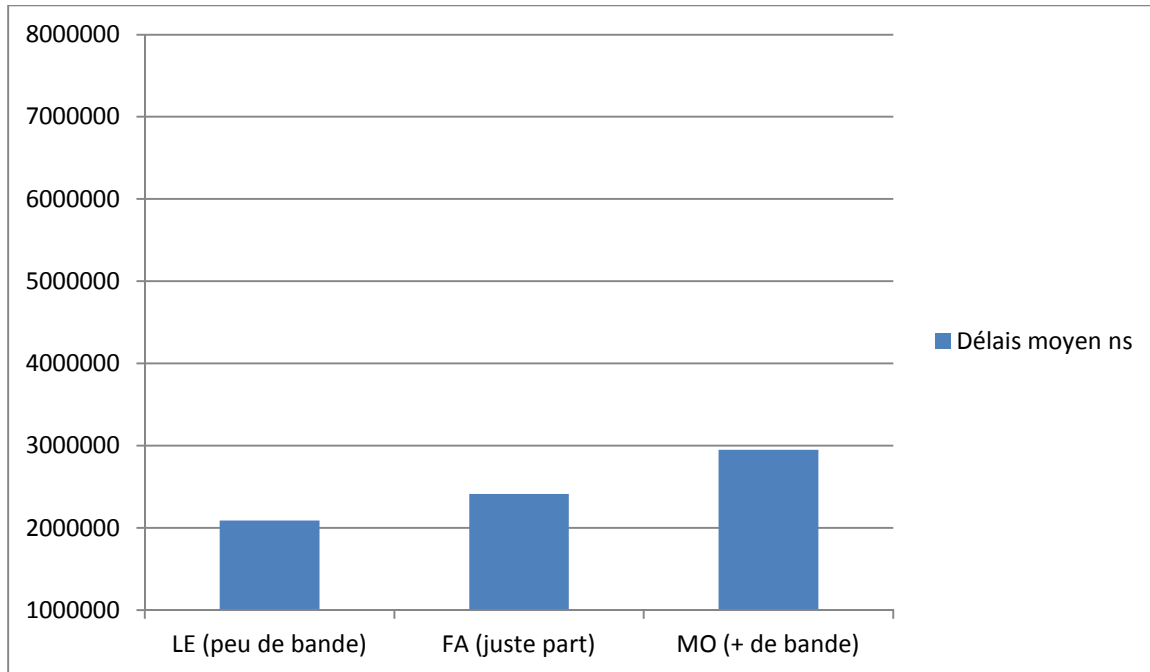


Figure 31 délais par type de flux avec QoS et équité sans transmissions conditionnelles

On remarque que les délais croissent avec les types de flux seulement quand l'équité est activée. Les algorithmes BSTFQ et FDCF rendent les délais des flux qui utilisent peu de bande passante mieux que ceux des flux qui utilisent leur juste part ou plus que leur juste part de bande passante.

5.2.2 Résultats avec l'algorithme des transmissions conditionnelles

Pour la seconde série d'expérimentations, nous reprenons les mêmes scénarios exécutés précédemment sans l'algorithme des transmissions conditionnelles, cette fois avec l'algorithme des transmissions conditionnelles activé.

5.2.2.1 Délais par classes de service sans QoS

Classe	Paquets Arrivés	Paquets Perdus	Délais moyen ns
VO (voix)	5867578	8166517	28399046
VI (vidéo)	6935942	9756594	31949658
BE (meilleur effort)	7109939	9953390	32225781
BK (arrière-plan)	7436980	10874395	32817514

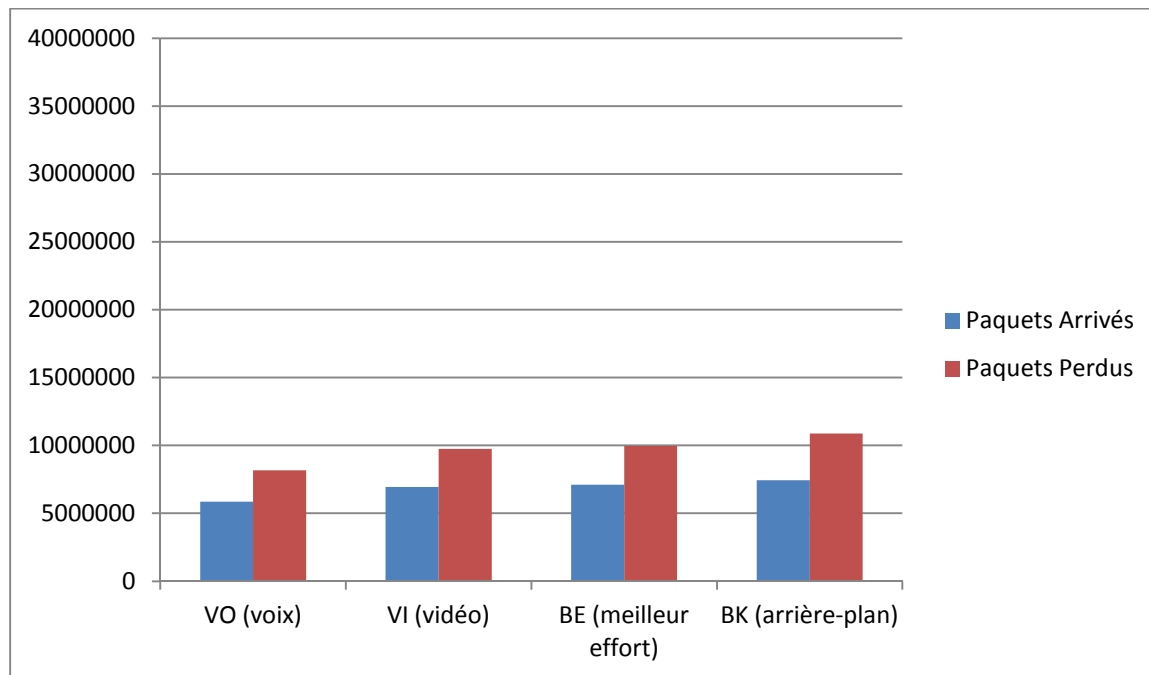


Figure 32 paquets par classe de service sans QoS avec transmissions conditionnelles

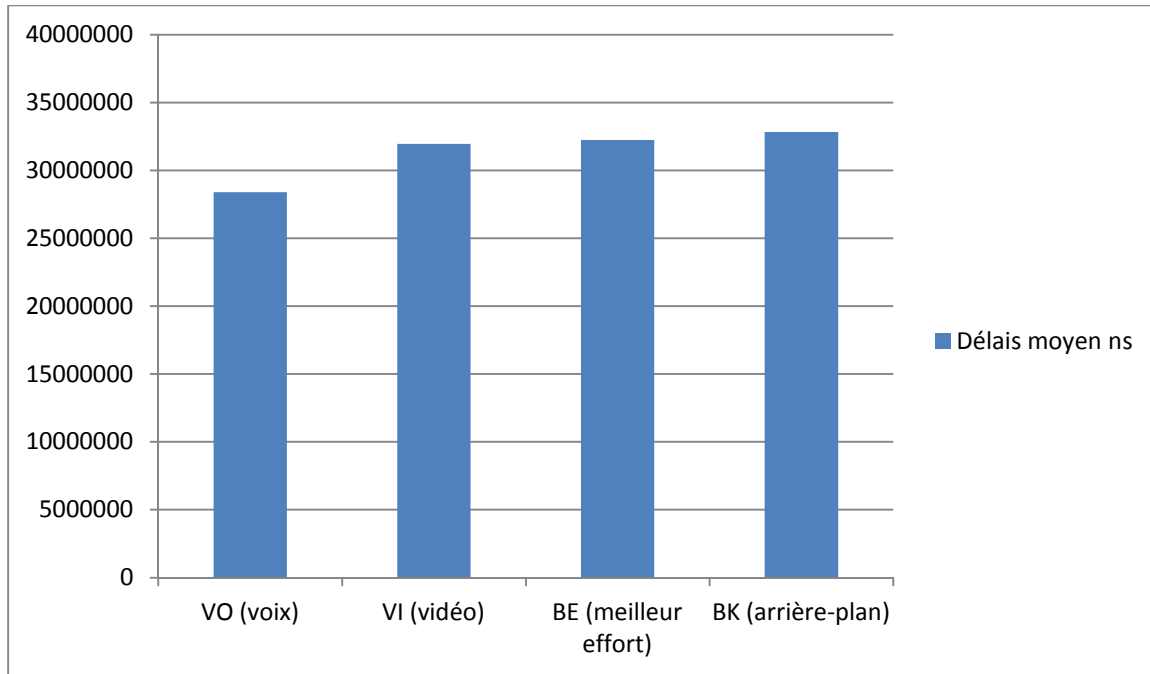


Figure 33 délais par classe de service sans QoS avec transmissions conditionnelles

Les délais sont relativement stables, voir le graphique avec QoS qui suit (5.2.2.2) pour constater la différence.

5.2.2.2 Délais par classe de service avec QoS

Classe	Paquets Arrivés	Paquets Perdus	Délais moyen ns
VO (voix)	2841635	3366767	4320467
VI (vidéo)	3523311	4627870	14168996
BE (meilleur effort)	6495518	9018175	30763107
BK (arrière-plan)	7076931	10150578	32421200

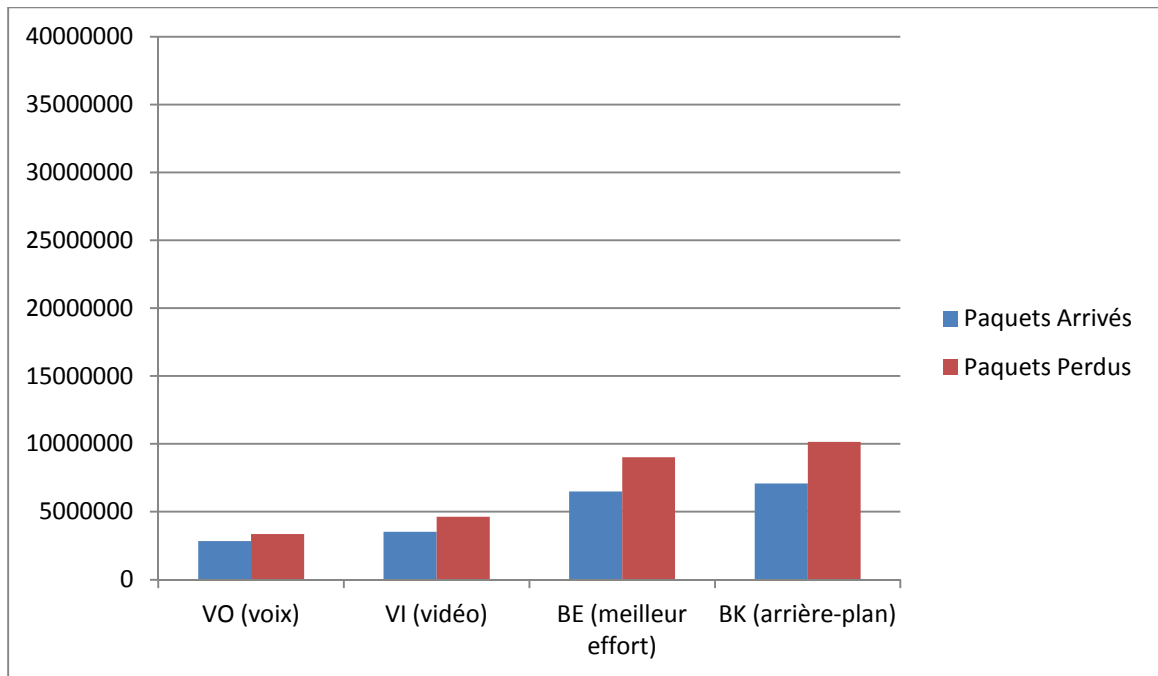


Figure 34 paquets par classe de service avec QoS avec transmissions conditionnelles

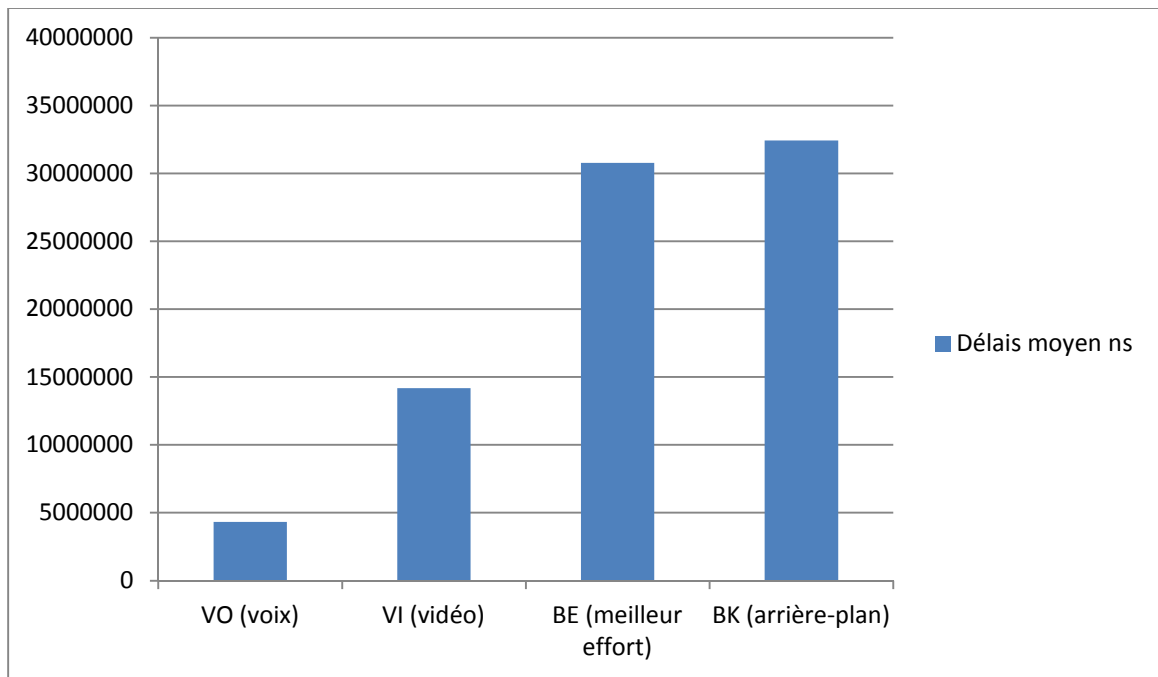


Figure 35 délais par classe de service avec QoS avec transmissions conditionnelles

Les délais croissent, les classes de service à gauche du graphique, respectivement VO et VI sont plus prioritaires que celles à droite du graphique, respectivement BE et BK. L'algorithme des transmissions conditionnelles affecte quand même les délais, le réseau est plus congestionné par son ajout, donc plus de paquets perdus.

5.2.2.3 Délais par classes de service avec QoS et Équité

Classe	Paquets Arrivés	Paquets Perdus	Délais moyen ns
VO (voix)	3313572	4468276	15955860
VI (vidéo)	4323472	5999561	24839466
BE (meilleur effort)	10877818	14835084	40201411
BK (arrière-plan)	10545474	15109213	40081616

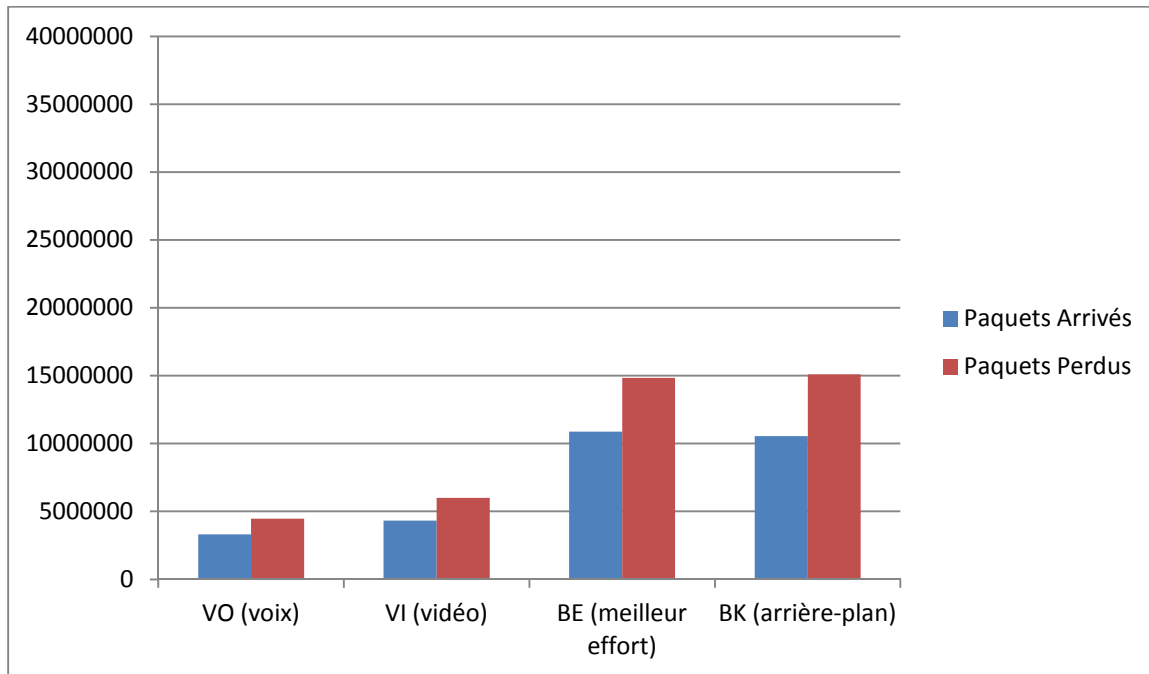


Figure 36 paquets par classe de service avec QoS et équité avec transmissions conditionnelles

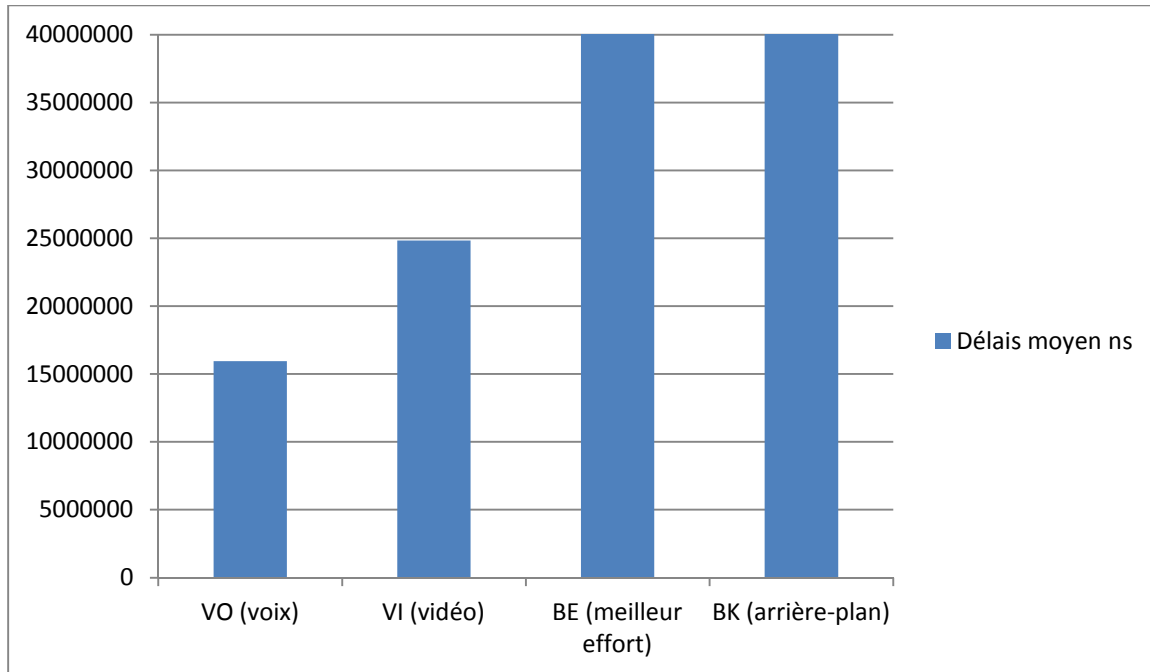


Figure 37 délais par classe de service avec QoS et équité avec transmissions conditionnelles

En activant les algorithmes FDCF et BSTFQ pour promouvoir l'équité, on voit que les délais sont toujours ordonnés, malgré que les deux classes de service à droite du graphique, celles qui sont moins prioritaires, reçoivent plus ou moins le même traitement, ce qui est dû à nos choix pour le contrôle de la fenêtre de contention tel que défini dans notre algorithme de la FDCF.

5.2.2.4 Délais par type de flux sans QoS

Type	Paquets Arrivés	Paquets Perdus	Délais moyen ns
LE (peu de bande)	1500133	1616655	4796484
FA (juste part)	2350186	2526916	3542860
MO (+ de bande)	23500120	34607325	35995535

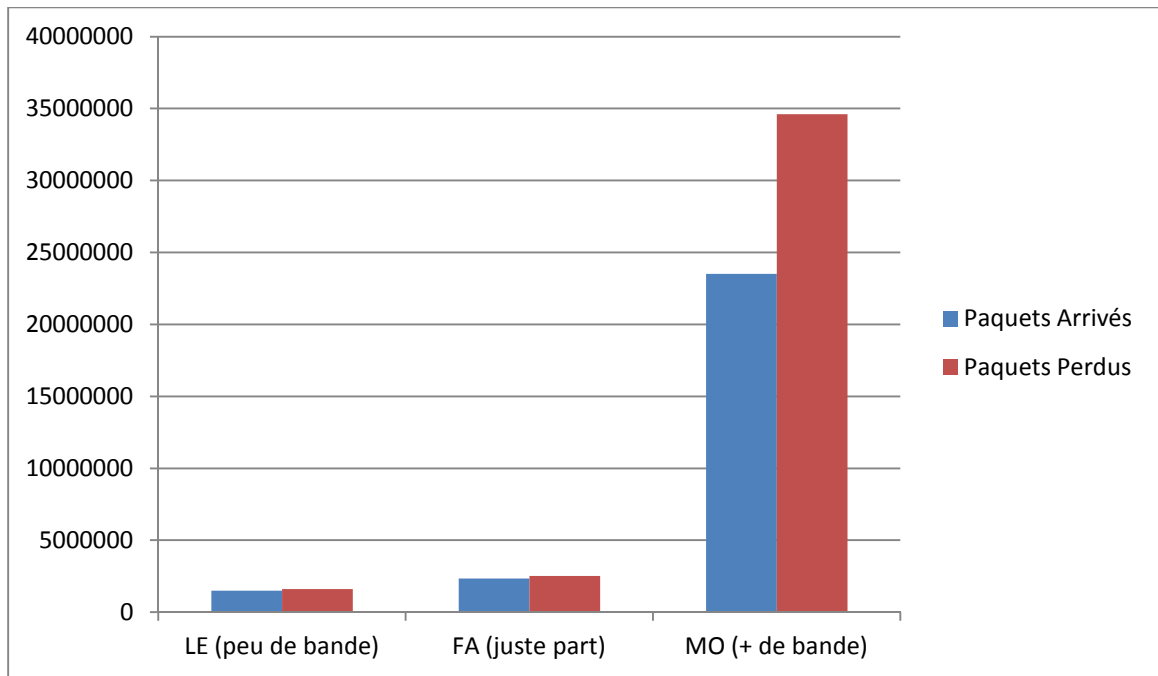


Figure 38 paquets par type de flux sans QoS avec transmissions conditionnelles

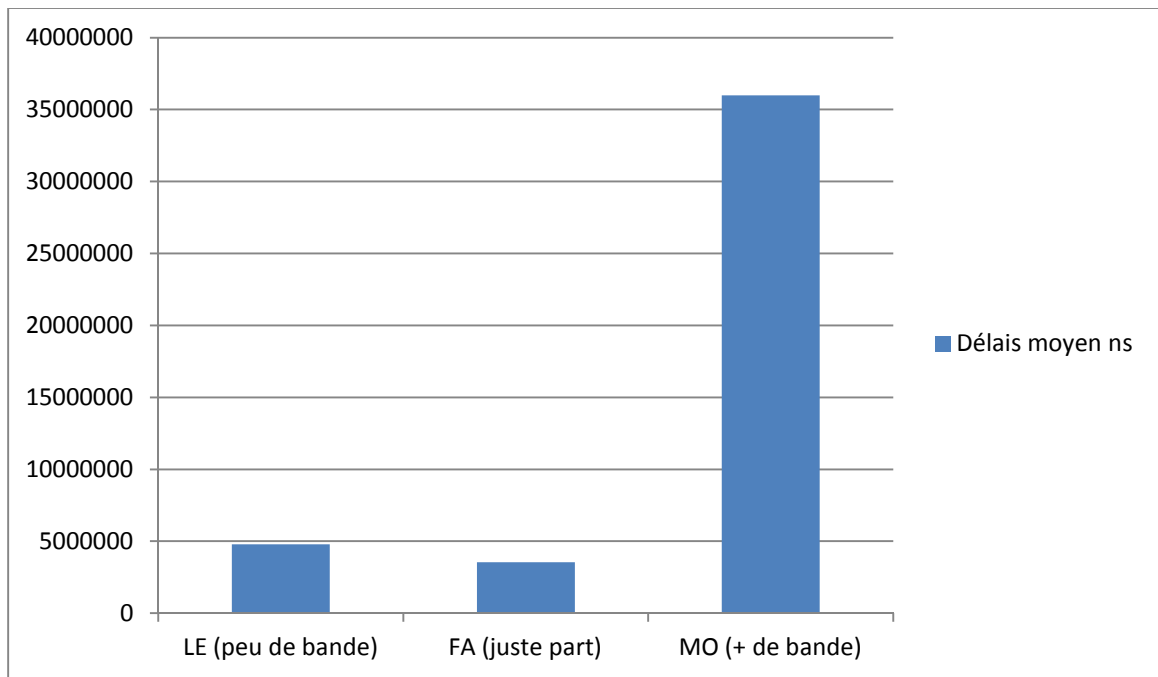


Figure 39 délais par type de flux sans QoS avec transmissions conditionnelles

Sans l'activation de la QoS, on remarque que les flux utilisant peu de bande passante sont désavantagés par rapport aux flux qui utilisent leur juste part de bande passante, et ce même si les flux qui utilisent plus que leur juste part de bande passante sont aussi désavantagés par rapport à tous les autres.

5.2.2.5 Délais par type de flux avec QoS

Type	Paquets Arrivés	Paquets Perdus	Délais moyen ns
LE (peu de bande)	1490815	1604413	3093504
FA (juste part)	2341449	2518639	2817237
MO (+ de bande)	16105131	23040338	29820038

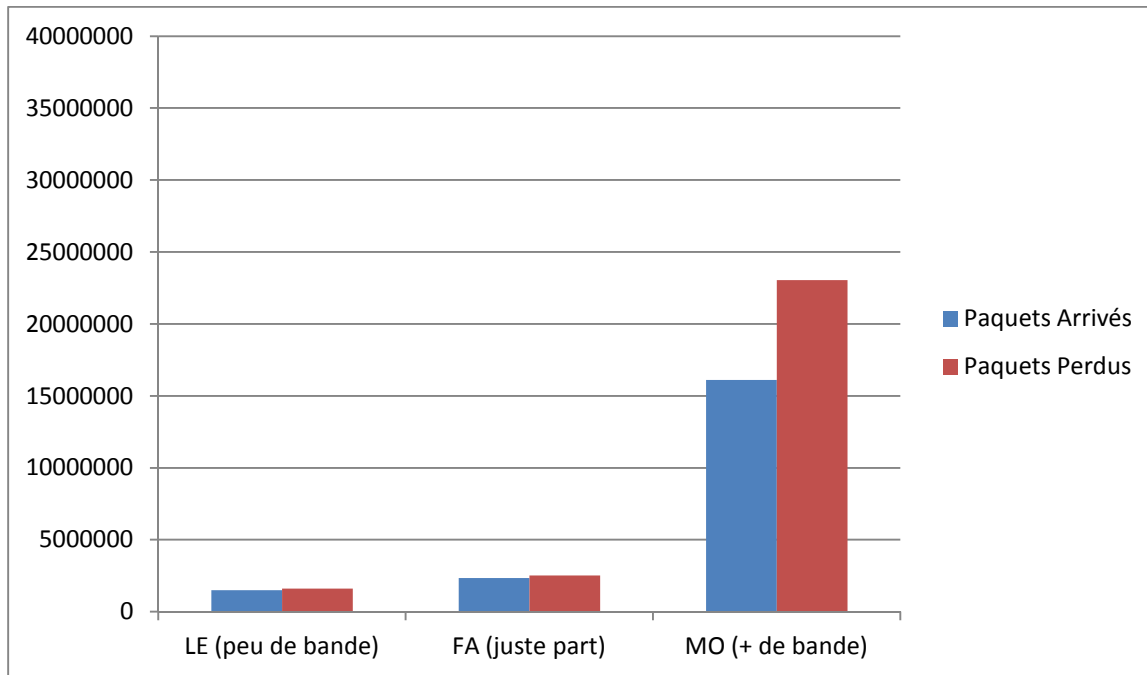


Figure 40 paquets par type de flux avec QoS avec transmissions conditionnelles

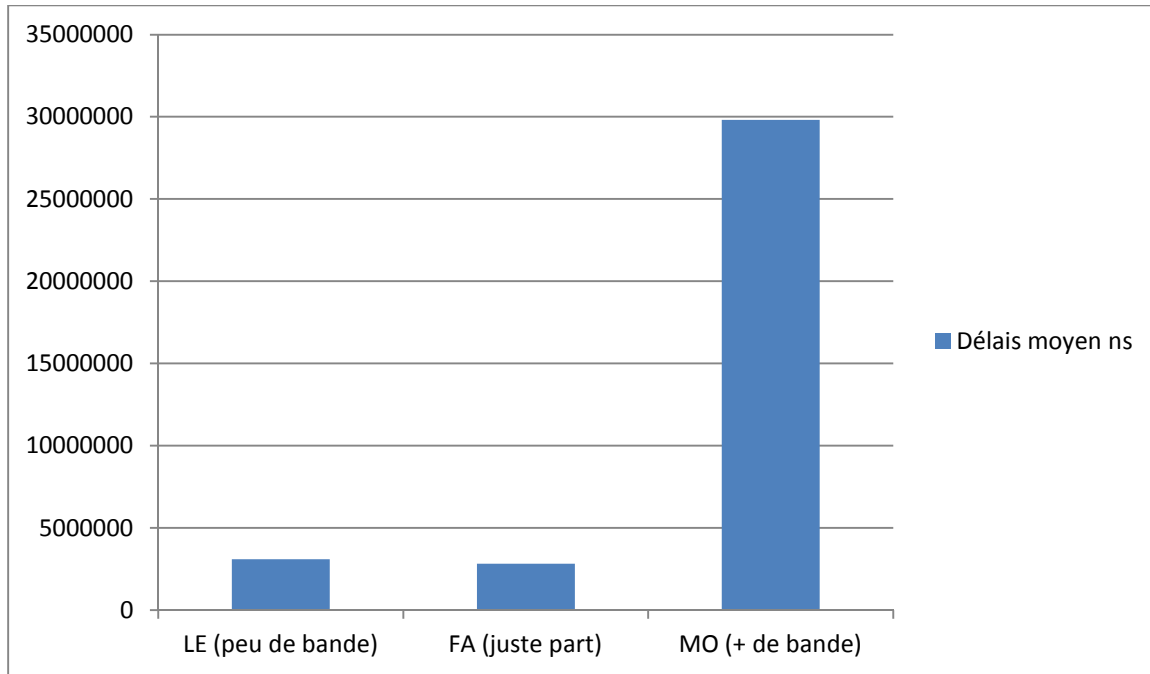


Figure 41 délais par type de flux avec QoS avec transmissions conditionnelles

Avec la QoS activée, peu de différence avec le graphique précédent.

5.2.2.6 Délais par type de flux avec QoS et Équité

Type	Paquets Arrivés	Paquets Perdus	Délais moyen ns
LE (peu de bande)	1282247	1406771	2124771
FA (juste part)	2025598	2222760	2481761
MO (+ de bande)	25752491	36782603	39316405

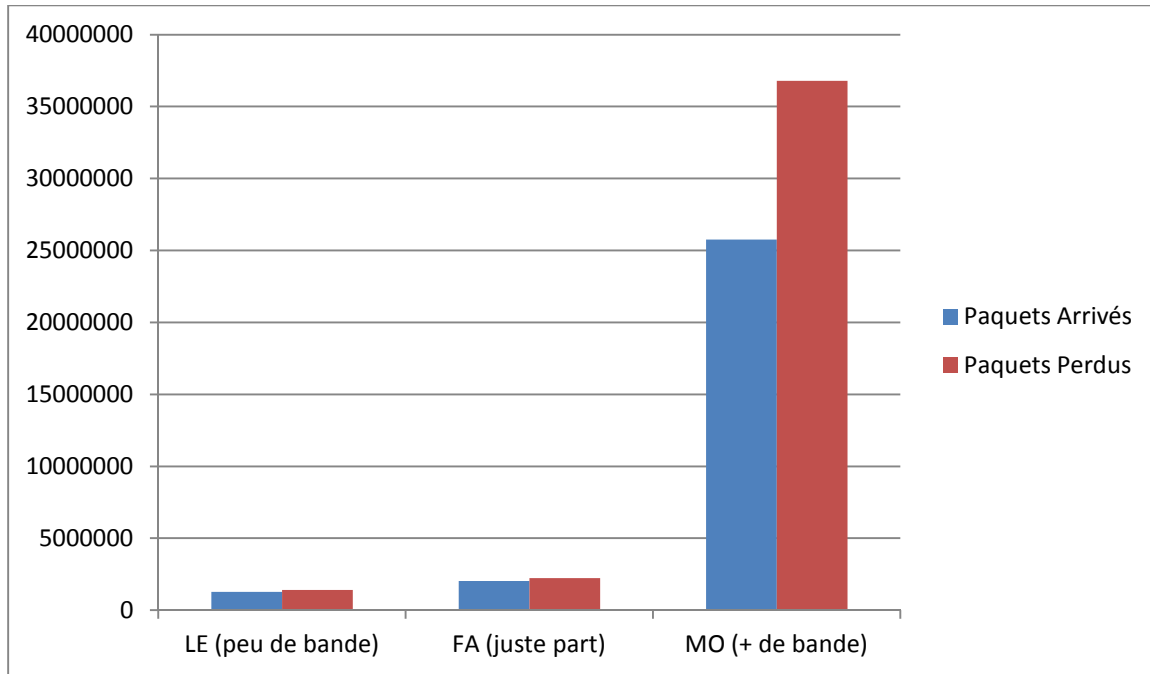


Figure 42 paquets par type de flux avec QoS et équité avec transmissions conditionnelles

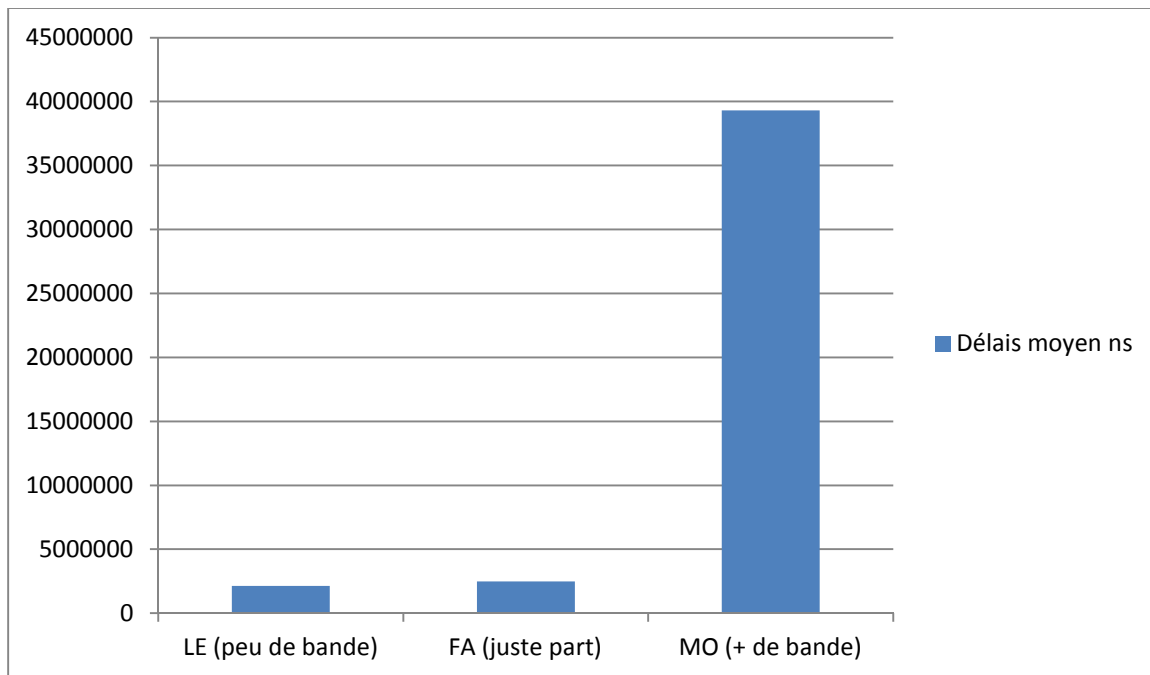


Figure 43 délais par type de flux avec QoS et équité avec transmissions conditionnelles

Une fois les algorithmes FDCF et BSTFQ qui font la promotion de l'équité activés, les flux qui utilisent peu de bande passante sont tout de même un peu avantagés par rapport aux autres, même si la marge avec les flux qui utilisent leur juste part de bande passante n'est pas énorme. La différence est quand même assez, vu le nombre réduit de paquets, pour influencer le comportement des conducteurs, comme on peut voir dans le tableau qui suit.

5.2.2.7 Influence sur décision des conducteurs selon la configuration

Véhicules	DéTECTÉS	DÉCIDÉS	Contournés	Préservés
Sans QoS	823	107	45	62
Avec QoS	822	99	42	57
QoS + Équité	823	124	41	83

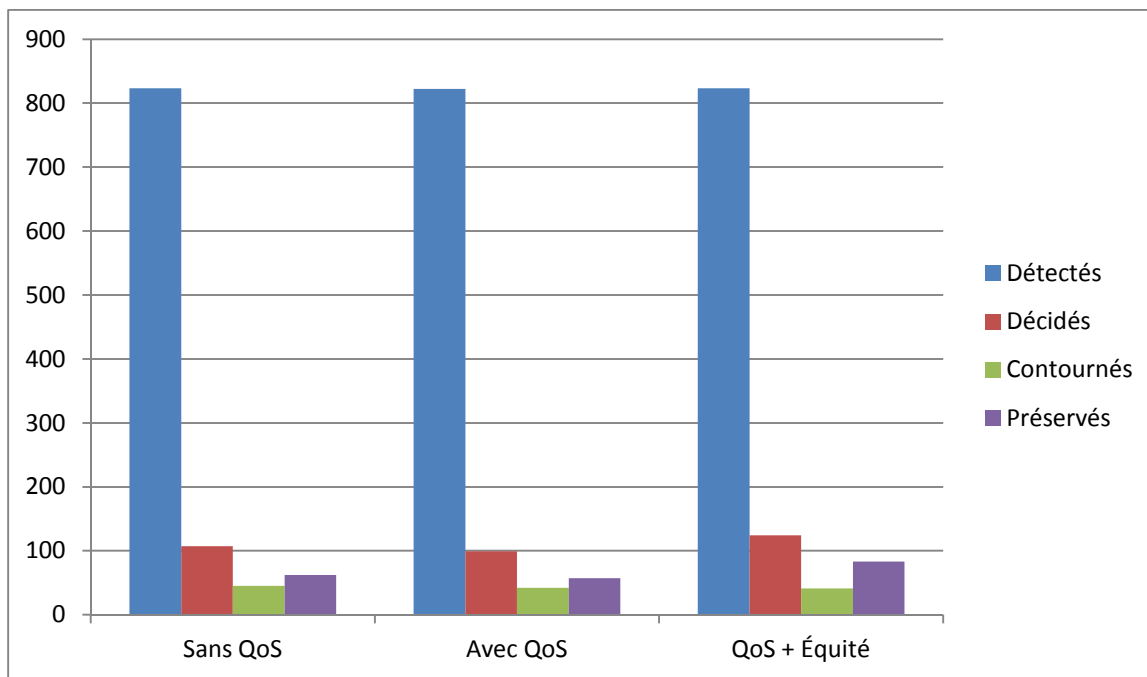


Figure 44 influence sur les décisions des conducteurs

Ce tableau montre que le nombre de véhicules qui ont détecté le bouchon est assez stable peu importe le sous-scénario. Ces véhicules ont détecté la condition de bouchon simplement en ralentissant leur vitesse, comportement modélisé par l'IDM. Par contre le nombre de véhicules

qui ont eu la chance de prendre une décision avant la sortie à été influencée par les communications provenant de l'application de détection de bouchon qui propage l'information avec l'algorithme de routage des transmissions conditionnelles. On remarque que le nombre de véhicule qui ont pu prendre une décision en obtenant cette information diminue avec la QoS. Ceci peut s'expliquer avec l'augmentation d'utilisation de bande passante causée par l'utilisation de la QoS. Cette augmentation est la même avec l'utilisation de l'équité avec les algorithmes FDCF et BSTFQ, mais est en fait optimisée par le fait que les flux sont triés selon leur usage de la bande passante, donc on peut dire que dans ce scénario de sortie d'autoroute pour éviter un bouchon, l'utilisation de nos algorithmes optimise la qualité de service et l'équité. En fait ce qui est important dans ce tableau c'est qu'on démontre que l'utilisation de l'équité peut augmenter les chances que l'information qui aide les conducteurs à prendre une décision se rendent à ceux-ci dans des situations de congestion. L'algorithme de détection de bouchon change la probabilité des véhicules de prendre la sortie une fois qu'ils ont reçu le message de détection de bouchon. Cette probabilité change de 80% qui vont tout droit et 20% qui prennent la sortie à 60% qui vont tout droit et 40% qui prennent la sortie. Cette différence de probabilité provient des travaux [42] sur les comportements des conducteurs.

6 Conclusion

Dans ces travaux, nous avons modifié les algorithmes EDCF et TFQ pour créer les algorithmes faisant la promotion de l'équité, FDCF et BSTFQ. La contribution proposée par nos travaux répond à nos objectifs de recherche, c'est-à-dire premièrement de modifier et intégrer des algorithmes de l'état de l'art pour valider que l'équité fournie par nos algorithmes FDCF et BSTFQ est un concept utile aux réseaux VANET dans un contexte qui met de l'avant les flux plutôt que les véhicules. Deuxièmement, nos travaux valident que ces modifications continuent de supporter la qualité de service, c'est-à-dire que celle-ci n'est pas affectée et l'observation des délais des différentes classes de service le confirme. Finalement, nos travaux valident que ces modifications optimisent l'équité et la qualité de service dans notre environnement d'expérimentation et son scénario. Des travaux futurs pourraient utiliser des techniques d'intelligence artificielle comme les radios cognitives, les véhicules pourraient même apprendre comment mieux gérer le retard local et chez les pairs même s'ils n'ont qu'une information partielle des pairs, sachant qu'on discute seulement de la possibilité dans le contexte de ces travaux pour garder la porte ouverte à ces techniques en utilisant la même infrastructure et les algorithmes ajoutés. Plus d'expérimentation serait nécessaire pour caractériser l'utilité de l'équité comme solution pour la gestion de la congestion, un concept complémentaire à la qualité de service. L'ajout du concept dans les standards de l'IEEE pour DSRC ou les protocoles du futur serait bénéfique aux communications VANET.

7 Références

- [1] Z. Jing and S. Roy, "MAC for dedicated short range communications in intelligent transport system," *Communications Magazine, IEEE*, vol. 41, pp. 60-67, 2003.
- [2] D. Jiang and L. Delgrossi, "IEEE 802.11p: Towards an International Standard for Wireless Access in Vehicular Environments," in *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, 2008, pp. 2036-2040.
- [3] X. Xin and N. ZhiSheng, "Performance of EDCF MAC scheme for future multi-service DSRC based road-to-vehicle communication systems in ITS," in *Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004. 15th IEEE International Symposium on*, 2004, pp. 2830-2834 Vol.4.
- [4] A. Demers, S. Keshav, and S. Shenker, "Analysis and simulation of a fair queueing algorithm," 1989, pp. 1-12.
- [5] M. J. Booyesen, S. Zeadally, and G. J. van Rooyen, "Survey of media access control protocols for vehicular ad hoc networks," *IET Communications*, vol. 5, pp. 1619-1631, 2011.
- [6] J. W. Ding, C. F. Wang, F. H. Meng, and T. Y. Wu, "Real-time vehicle route guidance using vehicle-to-vehicle communication," *Communications, IET*, vol. 4, pp. 870-883, 2010.
- [7] B. Ducourthial, Y. Khaled, and M. Shawky, "Conditional transmissions, a strategy for highly dynamic vehicular ad hoc networks," in *World of Wireless, Mobile and Multimedia Networks, 2007. WoWMoM 2007. IEEE International Symposium on a*, 2007, pp. 1-8.
- [8] M. S. Bouassida and M. Shawky, "On the congestion control within VANET," in *Wireless Days, 2008. WD '08. 1st IFIP*, 2008, pp. 1-5.

- [9] I. Mahadevan and K. M. Sivalingam, "Quality of Service architectures for wireless networks: IntServ and DiffServ models," in *Parallel Architectures, Algorithms, and Networks, 1999. (I-SPAN '99) Proceedings. Fourth International Symposium on*, 1999, pp. 420-425.
- [10] B. R. Smith and J. J. Garcia-Luna-Aceves, "Best Effort Quality-of-Service," in *Computer Communications and Networks, 2008. ICCCN '08. Proceedings of 17th International Conference on*, 2008, pp. 1-6.
- [11] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks-the single node case," in *INFOCOM '92. Eleventh Annual Joint Conference of the IEEE Computer and Communications Societies, IEEE*, 1992, pp. 915-924 vol.2.
- [12] T. Gyires, "An extension of Integrated Services with active networking for providing quality of service in networks with long-range dependent traffic," in *Systems, Man, and Cybernetics, 2001 IEEE International Conference on*, 2001, pp. 12-17 vol.1.
- [13] L. S. Brakmo and L. L. Peterson, "TCP Vegas: end to end congestion avoidance on a global internet," *IEEE Journal on Selected Areas in Communications*, vol. 13, pp. 1465-1480, 1995.
- [14] S. Y. Ameen and I. A. Ibrahimi, "MANET Routing Protocols Performance Evaluation with TCP Tahoe, Reno and New-Reno," *International Journal of U- & E-Service, Science & Technology*, vol. 4, pp. 37-49, 2011.
- [15] K. Kusume and G. Bauch, "CDMA and IDMA: Iterative Multiuser Detections for Near-Far Asynchronous Communications," in *Personal, Indoor and Mobile Radio Communications, 2005. PIMRC 2005. IEEE 16th International Symposium on*, 2005, pp. 426-431.
- [16] M. Shreedhar and G. Varghese, "Efficient fair queuing using deficit round-robin," *IEEE/ACM Transactions on Networking*, vol. 4, pp. 375-385, 1996.

- [17] H. T. Cheng, H. Shan, and W. Zhuang, "Infotainment and road safety service support in vehicular networking: From a communication perspective," *Mechanical Systems and Signal Processing*, vol. 25, pp. 2020-2038, 2011.
- [18] E. Karamad and F. Ashtiani, "A modified 802.11-based MAC scheme to assure fair access for vehicle-to-roadside communications," *Computer Communications*, vol. 31, pp. 2898-2906, 2008.
- [19] F. Harivelo, Anelli, P., "Equité pour réseau ad hoc Wi-fi," in *JDIR*, Marne la Vallée, France, 2007.
- [20] L. S. Y. Zang, X. Cheng, H.-J. Reumerman, A. Paruzel, and A. Barroso, "Congestion control in wireless networks for vehicular safety applications," in *8th European Wireless Conference*, Paris, France, 2007, p. 7.
- [21] T. P. Lee and G. Mercankosk, "Deficit Round Robin favors longer documents," in *TENCON 2006. 2006 IEEE Region 10 Conference*, 2006, pp. 1-4.
- [22] Z. Dwekat and G. N. Rouskas, "A practical fair queuing scheduler: Simplification through quantization," *Computer Networks*, vol. 55, pp. 2392-2406, 2011.
- [23] K. Hakyong, Y. Changmo, and J. Woo-Young, "Simulation study on the effect of the trTCM parameters," in *Telecommunications, 2003. ICT 2003. 10th International Conference on*, 2003, pp. 1482-1488 vol.2.
- [24] C. Bouras and A. Sevasti, "Performance enhancement of an AF service using TCP-aware marking and dynamic WRED," in *Computers and Communications, 2005. ISCC 2005. Proceedings. 10th IEEE Symposium on*, 2005, pp. 642-647.
- [25] B. Ducourthial, Y. Khaled, and M. Shawky, "Conditional Transmissions: Performance Study of a New Communication Strategy in VANET," *Vehicular Technology, IEEE Transactions on*, vol. 56, pp. 3348-3357, 2007.

- [26] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *Networking, IEEE/ACM Transactions on*, vol. 1, pp. 397-413, 1993.
- [27] N. Matsuo, M. Yuito, and Y. Tokunaga, "PACKET INTERLEAVING FOR REDUCING SPEECH QUALITY DEGRADATION IN PACKET VOICE COMMUNICATIONS," 1987, pp. 1787-1791.
- [28] M. Wada, "Selective recovery of video packet loss using error concealment," *IEEE Journal on Selected Areas in Communications*, vol. 7, pp. 807-814, 1989.
- [29] J. W. Ding, C. F. Wang, F. H. Meng, and T. Y. Wu, "Real-time vehicle route guidance using vehicle-to-vehicle communication," *IET Communications*, vol. 4, pp. 870-883, 2010.
- [30] F. Ducatelle, G. Di Caro, and L. M. Gambardella, "Ant agents for hybrid multipath routing in mobile ad hoc networks," in *Wireless On-demand Network Systems and Services, 2005. WONS 2005. Second Annual Conference on*, 2005, pp. 44-53.
- [31] C. Shen, D. Pesch, and J. Irvine, "A framework for self-management of hybrid wireless networks using autonomic computing principles," in *Communication Networks and Services Research Conference, 2005. Proceedings of the 3rd Annual*, 2005, pp. 261-266.
- [32] D. S. Damir Zrno, Ramjee Prasad, "Optimizing Cognitive Ad-Hoc Wireless Networks for Green Communications," *Journal of Green Engineering*, pp. 209-227, 2011.
- [33] S. B. Nazim Agoulmine, Dmitri Botvitch, and E. L. John Strassner, William Donnelly, "Challenges for Autonomic Network Management," presented at the MACE 2006, 2006.
- [34] M. Lee, D. Marconett, X. Ye, and S. J. Ben Yoo, "Cognitive network management with reinforcement learning for wireless mesh networks," vol. 4786 LNCS, ed, 2007, pp. 168-179.
- [35] L. Xiaojun, N. B. Shroff, and R. Srikant, "A tutorial on cross-layer optimization in wireless networks," *Selected Areas in Communications, IEEE Journal on*, vol. 24, pp. 1452-1463, 2006.

- [36] Z. Yang and C. Guohong, "V-PADA: Vehicle-Platoon-Aware Data Access in VANETs," *Vehicular Technology, IEEE Transactions on*, vol. 60, pp. 2326-2339, 2011.
- [37] H. A. a. M. C. Weigle, "Highway Mobility and Vehicular Ad-Hoc Networks in ns-3," in *Proceedings of the Winter Simulation Conference*, Baltimore, MD, 2010.
- [38] M. Treiber, A. Hennecke, and D. Helbing, "Congested Traffic States in Empirical Observations and Microscopic Simulations," *Physical Review E*, vol. 62, pp. 1805-1824, 2000.
- [39] S. R. Henderson T., S. Floyd, and G. F. Riley, "ns-3 Project Goals," in *Proceedings of the 2006 workshop on ns-2: the IP network simulator*, 2006.
- [40] (September 24). ns-3. Available: <http://www.nsnam.org/>
- [41] (September 24). *Highway Mobility for Vehicular Networks (VANET in ns-3)*. Available: <http://code.google.com/p/ns-3-highway-mobility/>
- [42] F. Dressler and C. Sommer, "On the Impact of Human Driver Behavior on Intelligent Transportation Systems," in *Vehicular Technology Conference (VTC 2010-Spring)*, 2010 IEEE 71st, 2010, pp. 1-5.

Annexe A syntaxe du pseudo-code

La syntaxe utilisée pour écrire notre pseudo-code s'apparente à la syntaxe du langage C. Notre syntaxe utilise donc les mêmes opérateurs, et les mêmes délimiteurs que le langage C, à l'exception qu'on n'utilise pas le point-virgule (;) pour terminer les énoncés, afin de simplifier. La structure d'une fonction sera donc comme suit.

```
int exemple (int premier_parametre, int deuxieme_parametre )  
  
{  
  
    const float pi = 3.1416  
  
    int rayon  
  
    ...  
  
    // commentaries  
  
    /* d'autres commentaries */  
  
    return ( 0 )  
  
}
```

Les mots clés ici sont int, const, float et return. Le mot clé int sert à définir un type, il est suivi d'un identificateur, dans ce cas-ci exemple, premier_parametre, deuxieme_parametre et rayon. L'identificateur exemple, servira à faire appel à la fonction dans le code. Le mot clé const sert à définir des identificateurs de constantes qui seront utilisées dans le code. Les accolades { et }, servent à délimiter le corps d'une fonction. Les // et /* */ servent à délimiter les commentaires, avec // pour les commentaires d'une seule ligne et /* pour les commentaires qui font possiblement plus de lignes, terminées par */.