Republique Algerienne Democratique et Populaire
Ministère de l'enseignement superieur et de la recherche scientifique
Universite Abderrahmane Mira, Béjaia
Ecole doctorale d'informatique
Reseaux et Systemes Distribués(Resyd)

Mémoire de Magister
Option : Réseaux et Systemes Distribués
Theme :

Voix sur IP et Algorithmes de contrôle Adaptatifs dans les réseaux Ad hoc

Présenté par:

Mohamed HADDACHE

Devant le jury composé de:

Président: Ahmed BERBOUCHA, Maître de Conférences, UAMB, Algérie.
Rapporteur: Ahmed MEDDAHI, Maître de Conférences, Telecom Lille1, France.
Examinateurs: Ali MELIT, Maître de Conférences, Université de Jijel, Algérie.
Abdallah BOUKERRAM, Maître de Conférences, UFA, Sétif, Algérie.

Année universitaire 2006/2007

Résumé

La transmission de la voix sur IP (VOIP) ou la téléphonie Internet connue aussi sous le nom « la téléphonie sur IP » complémente et remplace peu à peu la téléphonie classique (RTC : réseau téléphonique commuté) à cause des avantages offerts par la VOIP devant RTC. Mais le transport et la gestion la voix dans un réseau IP (Internet) pose un certain nombre de problèmes liés aux variations de la gigue, aux rallongements des délais et aux pertes de paquets. Ces problèmes deviennent encore plus graves dans un contexte de réseaux dynamiques, comme dans le cas des réseaux ad hoc, où ces paramètres de qualité de service (QoS) doivent encore être mieux contrôlés et maîtrisés que dans le cas de l'Internet filaire. Plusieurs modèles de QoS ont été proposés pour pallier ces problèmes. Dans ce mémoire nous avons étudié des modèles qui offrent des solutions pour les réseaux filaires à savoir : les modèles de contrôles de débit, les méthodes d'évitement de congestion, la réservation des ressources et les modèles de différentiation du service. Ainsi que les modèles de QoS pour les réseaux ad hoc à savoir : FQMM, SWAN, INSIGNIA, routage avec QoS, le protocole MRTP. Nous avons proposé aussi une extension du protocole MRTP pour le support d'un mécanisme de contrôle d'admission qui assure une QoS de bout en bout en terme de délai.

Mots clés : Voix sur IP, Réseaux Ad Hoc, Protocole RTP, RTCP, UDP, AIMD, MRTP, MRTCP, protocole de routage, routage multichemins

Abstract

Transmission of the voice over IP (VOIP)or Internet telephony also known under the name "telephony over IP" complemente and replaces little by little traditional telephony (PSTN: public switched telephone network) this is for the advantages offered by the VOIP in front of PSTN. But the transport and the management of the voice in an IP network (Internet) pose a number of problems involved in the variations of the jitter, the extensions of the delay and the losses of packages. These problems become even more serious in a context of dynamic networks, as in the case of the Ad hoc networks, where these parameters of quality of service (QoS) must still be better controlled and controlled that in the case of the wired Internet. Several models of QoS were proposed to mitigate these problems. In this works we have studies models which offer solutions for the wired networks like: flow controls models, avoidance congestion methods, the reservation of the resources and the service differentiation models. Also, QoS for Ad hoc networks like: FQMM, SWAN, INSIGNIA, routing with QoS, MRTP. We also proposed architecture: MRTP with control of admission which ensures an end to end QoS in term of delay.

Keys Words: VOIP, Ad hoc networks, RTP protocol, RTCP,UDP, AIMD,MRTP, MRTCP, routing protocol,...

Dedicaces

A mes parents

A mes frères Kamel, Rachid, Belkacem et Sid Ali

A ma seule soeur Yasmina.

A toute ma famille maternelle et paternelle

A mes Amis

A mes étudiants

A tous je dedie ce modeste travail.

Remerciements

D'abord, je tiens à exprimer mes plus vifs remerciements à mon promoteur, monsieur MEDDAHI Ahmed pour les nombreux conseils, orientations et encouragements afin que ce travail soit à la mesure des exigences.

Je remercie les membres du jury d'avoir accepté de juger ce travail.

Je remercie également Mr TARI Kamel pour ses contributions à l'école doctorale (ReSyd).

Je désire également remercier mes collègues de l'école doctorale d'informatique pour leur soutien moral.

Mes vifs remerciements s'adressent aussi à tous les professeurs de l'intérieur et de l'extérieur de l'Algérie, qui ont contribué dans notre formation de première année magistère à l'école doctorale d'informatique de l'université Abderrahmane Mira de Béjaia.

Mes remerciements s'adressent aussi à tous mes étudiants, qui m'ont encouragé durant toute l'année

Table des matières

Liste des figures **ix**

Liste des tables **x**

Liste des algorithmes **xi**

Liste des abréviations **xii**

Introduction **1**

1 Voix sur IP et Qualité de service **4**

1.1 Introduction 4

1.1.1 Avantages de VOIP 5

1.2 Transmission de la voix sur IP 5

1.2.1 Les phases de transmission de la voix sur un réseau IP 6

1.2.2 La pile des protocoles Internet pour la transmission de la voix 7

1.2.3 Les protocoles de transmission de la voix sur IP 8

1.2.4 Les protocoles de signalisation 10

1.3 Les paramètres affectant la transmission de la voix sur IP 13

1.3.1 Qualité du codage 13

1.3.2 Délai de bout en bout 14

1.3.3 La gigue : 14

1.3.4 Perte des paquets 15

1.3.5 La largeur de la bande passante 15

1.4 Solutions pour améliorer la qualité de service 16

1.4.1 Le contrôle de congestion 16

1.4.2 Le contrôle de débit 18

1.4.3 Reservation des ressources 20

1.4.4 Differentiation de service 23

1.5 Conclusion 28

2 Les réseaux sans fil et le routage **30**

2.1 Introduction . . . 30
2.2 Architectures des réseaux sans fil . . . 30
2.2.1 L'architecture avec point d'accès . . . 30
2.2.2 L'architecture Ad hoc . . . 31
2.3 Les Applications des réseaux sans fil . . . 31
2.4 Normes des réseaux sans fil . . . 31
2.4.1 La norme 802.11 . . . 32
2.5 Les réseaux Ad Hoc . . . 34
2.5.1 Caractéristiques des réseaux Ad hoc . . . 34
2.6 Routage dans les réseaux Ad Hoc . . . 35
2.6.1 Spécificité de routage dans les réseaux sans fil . . . 36
2.6.2 Evaluation d'un protocole de routage pour les réseaux sans fil . . . 37
2.6.3 Les protocoles de routage dans les réseaux Ad hoc . . . 38
2.6.4 Influence de type de protocole sur la transmission de la voix . . . 49
2.6.5 Routage multichemins . . . 50
2.6.6 Les avantages de routage multichemins . . . 50
2.7 Conclusion . . . 54

3 Les modèles de qualité de service dans les réseaux Ad hoc **55**

3.1 Introduction . . . 55
3.2 Le model FQMM . . . 55
3.3 Le model SWAN . . . 57
3.4 Les protocoles de QoS avec signalisation . . . 59
3.4.1 Le protocole de réservation de ressources INSIGNIA . . . 59
3.5 Les protocoles du routage avec QoS . . . 63
3.5.1 Le protocole CEDAR . . . 64
3.5.2 Le protocole des étiquettes basé sur le sondage . . . 64
3.6 Conclusion . . . 65

4 Le protocole MRTP **67**

4.1 Introduction . . . 67
4.2 Le protocole MRTP . . . 67
4.3 Gestion des flux et Sessions . . . 69
4.4 Partitionnement du trafic . . . 69
4.5 Etablissement des routes pratiquement . . . 69
4.6 Horodateur (Timestamping) . . . 69
4.7 Rapports de QoS . . . 70
4.8 Réassemblage des paquets au niveau du récepteur . . . 70

4.9 Format des paquets MRTP/MRTCP 71
4.9.1 Les paquets de données 71
4.9.2 Les rapports de contrôle de QoS (MRTCP QoS Reports) 72
4.9.3 Les Messages de contrôle de session / flux 73
4.9.4 Extension de l'en-tête 75
4.10 Les opérations MRTP/MRTCP 76
4.10.1 Etablissement et fermeture d'une session MRTP 76
4.10.2 Transfert des données 77
4.10.3 Rapport de qualité de service 77
4.10.4 Gestion des flux 77
4.11 Conclusion 77

5 MRTP avec contrôle d'admission 79

5.1 Introduction 79
5.2 Proposition 80

5.2.1 Module classificateur 81
5.2.2 Contrôle d'admission 82
5.2.3 Le module mesure du delai 83
5.2.4 Relation entre le protocole MRTP/MRTCP et le module contrôle d'admission 89
5.2.5 Exemple de déroulement de la proposition 90
5.3 Conclusion 91

6 Conclusion Generale et perspectives 93

Liste des Figures

1.1 Les différentes étapes de transmission de la voix 6
1.2 La pile des protocoles Internet Multimedia 7
1.3 Relation entre RTP et RTCP 10
1.4 Architecture H.323 11
1.5 Architecture SIP 12
1.6 Distribution de délai et illustration de la gigue 15
1.7 Les principaux problèmes affectant la transmission de la voix sur IP 15
1.8 Slow Start 17
1.9 Influence de slow start sur le délai 17
1.10 Seau troué 19
1.11 Seau à jetons 20
1.12 Fonctionnement du protocole RSVP 22
1.13 Classification et ordonnancement des paquets dans DiffServ 23
1.14 Architecture DiffServ 24
1.15 Files prioritaires 25
1.16 Round robin 26
1.17 Weighted Round Robin 26

2.1 Illustration de système CSMA 33
2.2 Illustration du mode d'accès DCF 34
2.3 Exemple du reseaux Ad hoc [38] 34
2.4 Changement de la topologie dans les reseaux Ad Hoc 35
2.5 Inondation 38
2.6 La recherche d'une route 40
2.7 Mécanisme de récupération d'une route avec AODV 40
2.8 Découverte de la route dans DSR 42
2.9 Fonctionnement des protocoles proactifs 44
2.10 Les relais multipoints 45
2.11 Principe de fonctionnement de FSR 46
2.12 Principe de fonctionnement de protocole ZRP 47

2.13 Le protocole CBRP 48
2.14 Comparaison entre la transmission de la voix à travers AODV et OLSR en terme de débit 49
2.15 Comparaison entre la transmission de la voix à travers AODV et OLSR en terme de délai 50
2.16 Propagation de RREP dans le protocole TORA 52
2.17 Création des routes dans le protocole TORA 53
2.18 Réaction du protocole TORA à la rupture d'un lien 53

3.1 Exemple de FQMM 56
3.2 Le modèle SWAN 57
3.3 Illustration d'installation de flux[24] 61
3.4 Illustration d'une restauration 62
3.5 Position de INSIGNIA dans le modèle de gestion du flux dans les routeurs . 63

4.1 Session MRTP[55] 68
4.2 Position du protocole MRTP/MRTCP dans la pile TCP/IP[55] 68
4.3 Format du paquet MRTP de données [55] 71
4.4 Format de paquet emetteur de MRTP[55] 73
4.5 Format d'un paquet hello session [55] 74
4.6 Format de message Bye session.[55] 75
4.7 Format d'en-tête d'extension [55] 75
4.8 les differents opération MRTP [55] 76

5.1 MRTP avec contrôle d'admission 80
5.2 Système avec file d'attente à K paquets 88
5.3 Exemple d'un reseau Ad hoc 90

Liste des Tables

1.1 Format d'un en-tête RTP 8
1.2 Les différents types de rapport RTCP 9
1.3 Caractéristiques de quelques codeurs/decodeurs 14
1.4 Les délais recommandés par ITU-T pour la voix[4] 14

3.1 Constituants du paquet INSIGNIA 60

5.1 les valeurs possibles pour le champs payload de MRTP 81
5.2 Constituants du paquet de chemin1 90
5.3 Constituants du paquet de chemin2 91
5.4 Constituants du paquet de chemin3 91
5.5 Le de contenue de liste_chemin 91

Liste des algorithmes

.

3.1 Algorithme AIMD. 58

5.1 Algorithme classifieur. 82
5.2 Algorithme controle d'admission. 83
5.3 Algorithme QoS_recepteur. 85
5.4 Algorithme QoS_emetteur. 86

Liste des abréviations

AF:	Assured Forwarding.
AIMD	Additive Increase Multiplicative Decrease.
ATM:	Asynchronous Transfer Mode
AODV:	Ad hoc On Demand Distance Vector routing protocol.
BE	Best Effort.
BL	Base Level.
CBR:	Constant Bit Rate.
CBRP:	Cluster Based Routing Protocol
CCID	Congestion Control Identifier.
CEDAR:	Core Extraction Distributed Ad hoc Routing
CNAME	Canonical Name
CSMA/CA:	Carrier Sense Multiple Access with Collision Avoidance.
CTS:	Clear To Send.
CW	Contention Window.
DCCP	Datagram Congestion Control Protocol
DCF:	Distributed Coordination Function.
DAG	Directed Acyclic Graph
DHCP:	Dynamic Host Configuration Protocol
DiffServ:	Differentiated Services.
DRR	Deficit Round Roubin.
DS	Direct Sequence.
DSCP:	Differentiated Services code point
DSR:	Dynamic Source Routing protocol
DS-SWAN:	Differentiated Services-Stateless Wireless Ad hoc Networks
ECN	Explicit Congestion Notification
EF	Expedited Forwarding.
EL	Enhancement Level.
FA-SWAN:	Fast Admission-Stateless Wireless Ad hoc Networks

FEC:	Forward error correction
FH:	Frequency Hopping.
FIFO	First In First Out.
FQ	Fair Queuing
FQMM	Flexible Quality of service Models for Manets.
FSR	Fisheye State Routing.
FTP	File Transfer Protocol.
GL	Controlled Load.
GPS	General Processor Sharing.
GS	Guaranteed Service.
GSM	Global System for Mobile communications.
HTTP	Hyper Text Transfer Protocol.
HiperLan	High Performance Local Area Network
IARP	IntrAzone Routing Protocol.
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IERP	IntErzone Routing Protocol.
IETF	Internet Engineering Task Force
IFS	Inter Frame Space.
IntServ	Integrated Services.
ISDN	Integrated Service Data Network
IP	Internet Protocol.
ITU	International Telecommunication Union.
UIT-T	Union Internationale des Télécommunications . standardisations du secteur Télécommunications
IR	Intra Red.
LAN	Local Area Network
MAC	Medium Access Control
MANET	Mobile Ad hoc Network
MCU	Multipoint Control Unit

MPR	Multi Point Relays
MRTP	Multi-flow Real time Transport Protocol
MRTCP	Multi-flow Real time Transport Control Protocol
OLSR	Optimized Link State Routing
OSPF	Open Shortest Path First.
PCF	Point Coordination Function
PHP	Per Hop Behaviour
PPP	Point to Point Protocol.
PRNet	Packet Radio Network
PSTN	Public Switched Telephonie Network
QoS	Quality of Service.
RED	Random Early Detection
REQ	Request
RES	Reserved
RFC	Request For Comments
RR	Receiver Report.
Rr	Round robin
RREQ	Route Request
RREP	Route Reply.
RRER	Route Reply Erreur
RSVP	Resource Reservation Setup Protocol
RT	Real Time
RTCP	Real time Control Protocol
RTP	Real Time Transport Protocol
RTS	Request To Send.
RTP	Round Trip Time
SACK	Selective Acknowledgement

SDES	Source Description
SIP	Session Initiation Protocol
SMTP	Simple Mail Transfer Protocol
SPF	Shortest Path First.
SR	Sender Report.
STCP	Stream Control Transmission Protocol.
SURAN	Survivable Radio Networks.
SWAN	Service differentiation in Wireless Ad hoc Networks
TBRPF	Topology Broadcast based on Reverse-Path Forwarding
TC	Topology Control
TCP	Transmission Control Protocol.
TLS	Transmission Layer Security
TORA	Temporary Ordering Routing Algorithm.
TOS	Type of Service.
TTL	Time To Live
UDP	User Datagram Protocol
URL	Uniform Resource Locators.
VOIP	Voice Over Internet Protocol.
WAN	Wide Area Network
WRED	Weighted Random Early Detection.
WRR	Weighted Round Roubin.
WFQ	Weighted Fair Queuing
ZRP	Zone Routing Protocol

Introduction Generale

Depuis seize ans, les évolutions de l'informatique et des télécommunications ont conduit à une modification radicale du paysage de la télécommunication informatique, et en conséquence de l'Internet et de ses services [9]. Initialement basées sur l'échange de données textuelles (le transfert de fichiers, mail, etc.), les applications distribuées manipulent à présent tous les types de médias (en premier lieu l'audio et la vidéo...). En comparaison avec les applications classiques, les applications multimédias présentent de nouvelles contraintes sur le transfert de certains de leurs médias (plus spécifiquement l'audio et la vidéo), tel que le délai de bout en bout, la gigue (la variation du temps d'arrivée des paquets), les pertes des paquets et la bande passante

Actuellement parmi les applications très importantes sur Internet ou sur les réseaux sans fil, on trouve la transmission de la voix sur IP [22].Cette application a connu une croissance énorme ces dernières années. Par exemple dans U.S.A, Forrester Research a prédit que le marché VoIP a été développé de 30 millions de dollars en 1998 à un marché de 2 milliards de dollars en 2004 [23]. Cette évolution peut être justifiée par les avantages ouverts par le téléphone IP devant le RTC (réseau téléphonique commuté ou PSTN : public switched téléphone network). Du point de vue utilisateur, le téléphone IP permet la réduction des coûts de communication ainsi la transmission de la voix avec d'autres médias tels que la vidéo ou avec les données textuelles. Par contre, le RTC ne permet pas une grande flexibilité, car il repose sur des technologies propriétaires peu adaptées à d'autres utilisations que celles prévues au départ. D'autre part, le téléphone IP permet aussi aux opérateurs des réseaux de réduire la largeur de la bande passante par l'utilisation des techniques de compression de la voix.

La transmission de la voix sur IP est très difficile, car les réseaux IP tels qu'ils ont été conçus au départ ne permettent de fournir aucune qualité de service. Aujourd'hui avec l'apparition des réseaux sans fil et plus particulièrement les réseaux ad hoc où les nœuds sont libres de se déplacer aléatoirement et s'organisent arbitrairement, par conséquent, la topologie du réseau peut varier de façon rapide et surtout imprévisible. Ces réseaux jouent un rôle très important dû à leur bas coût d'implémentation comme ils peuvent être installés facilement dans le désert et peuvent résister en cas de catastrophes naturelles ou de guerres.

Par conséquent, garantir une qualité de service pour la transmission de la voix au-dessus de ces réseaux s'avère une tache plus délicate, car d'une part la voix présente des contraintes strictes sur le délai de bout en bout qui ne doit pas dépasser 150ms, le taux de perte qui ne doit pas dépassé 4 %,... d'une autre part la mobilité des nœuds de ces réseaux cause des ruptures des chemins ce qui augmente le taux de perte ainsi que le délai de bout en bout (recherche d'un nouvel itinéraire).

Plusieurs modèles ont été proposés pour améliorer la qualité de service dans les réseaux filaires, mais les deux modèles les plus utilisés sont IntServ qui assure une qualité de service par flux et DiffServ qui assure une QoS par classe de flux. Malgré les améliorations fournies par ces deux modèles pour les réseaux filaires, ils restent insuffisants, car ne peuvent pas fournir une bonne qualité de service pour les réseaux mobiles en particulier les réseaux Ad hoc, où les nœuds sont libres de se déplacer aléatoirement et s'organisent arbitrairement.

Plusieurs modèles ont été proposés pour améliorer la qualité de service dans les réseaux Ad hoc tels que (IMAC, SWAN, FQMM...).

Organisation du memoire

Notre mémoire est organisé en cinq chapitres comme suit :

Dans le premier chapitre, nous présentons les étapes nécessaires pour la transmission de la voix sur IP à savoir : la digitalisation, la compression, la paquetisation et la transmission. Ensuite, nous présentons les protocoles de signalisations utilisés pour la transmission de la voix. Après, nous exposons les différents problèmes affectant la transmission de la voix sur IP, suivis par les solutions proposées pour améliorer la QoS dans les réseaux filaires.

Le deuxième chapitre est consacré à l'étude des réseaux Ad hoc, nous nous intéressons en particulier à leurs caractéristiques et aux protocoles de routages utilisés dans ces derniers. Nous avons classé les protocoles de routage dans les réseaux Ad hoc en trois catégories : premièrement les protocoles réactifs, ces protocoles ne cherchent la route qu'a la demande par une application.parmi ces protocoles en trouve AODV, DSR....Deuxièmement les protocoles proactifs : au contraire des précédents, ces protocoles entretiennent toutes les routes du réseau par l'échange périodique des trames de contrôle. Donc, il est possible de fournir instantanément la route vers une destination. Troisièmement, les protocoles hybrides qui sont un mélange entre les deux précédents.

Dans le troisième chapitre, nous exposons les différents modèles de qualité de service dans les réseaux Ad hoc tels que : FQMM, SWAN, INSIGNIA...

Dans le quatrième chapitre, nous étudions, le protocole MRTP/MRTCP (Multi-flow Realtime Transport Protocol/ Multi-flow Realtime Transport control Protocol) est un protocole de transport des flux multimédias, qui utilise des chemins multiples et ainsi que la technique de rétroaction.

Dans le cinquième chapitre, nous avons proposé une nouvelle architecture pour améliorer la qualité de service pour les flux voix sur les réseaux Ad hoc. Cette architecture est basée sur le routage multichemins et le mécanisme feedback utilisé par le protocole MRTP/MRTCP.

Enfin, notre mémoire s'achève par une conclusion générale résumant les grands points qui ont été abordés dans ce mémoire, ainsi que des perspectives que nous souhaitons accomplir prochainement

1

Voix sur IP et Qualité de service

1.1 Introduction

La transmission de la voix sur IP ou la téléphonie Internet connue aussi sous le nom « la téléphonie sur IP » complémente et remplace peu à peu la téléphonie classique (RTC : réseau téléphonique commuté). La téléphonie Internet est la livraison en temps réel de la voix et d'autres types de données et de multimédia entre deux nœuds ou plus à travers le réseau, en utilisant les protocoles Internet. Plus précisément, la transmission de la voix sur IP consiste à découper le signal numérique obtenu par numérisation de la voix en paquets qui seront transmis ensuite sur un réseau IP vers une application qui se chargera de la transformation inverse (des paquets vers la voix).

Les laboratoires de recherche, les entreprises et les universités gèrent deux réseaux séparés : un réseau de données, concrètement IP typiquement sur Ethernet, et un réseau téléphonique en étoile autour d'un autocommutateur, PABX (Private Automatic Branch Exchange). Ces deux réseaux sont généralement administrés par deux équipes différentes. Le réseau téléphonique, avec une facture en moyenne dix fois supérieure à celle de la transmission de données, est un poste budgétaire très important qui malgré la baisse du coût des communications n'a pas tendance à chuter de manière très significative. Ces deux réseaux peuvent (pour les plus récents) utiliser le même câblage constitué de paires torsadées avec les mêmes répartiteurs.

Actuellement au lieu de disposer à la fois d'un réseau informatique de données et d'un réseau téléphonique commuté (RTC), les entreprises peuvent donc, grâce à la VoIP, tout fusionner sur un même réseau. La VoIP doit non seulement simplifier le travail, mais aussi faire économiser de l'argent. Les entreprises dépensent énormément en communications téléphoniques, or le prix des communications de la Toip (Téléphonie sur IP) est dérisoire en comparaison. En particulier, plus les interlocuteurs sont éloignés, plus la différence de prix est intéressante. De plus, la téléphonie sur IP utilise jusqu'à dix fois moins de bande passante que la téléphonie traditionnelle. Ceci apportant de grands intérêts pour la voix sur le réseau privé.

1.1.1 Avantages de VOIP

La VoIP offre de nombreuses nouvelles possibilités aux opérateurs et utilisateurs qui bénéficient d'un réseau basé sur IP. Les avantages les plus marqués sont les suivants :

Réduction des coûts

En déplaçant le trafic voix de RTC vers le réseau privé WAN/IP, les entreprises peuvent réduire sensiblement certains coûts de communications.

-Fonctionnement : la possibilité de ramener la facture téléphonique dans le même ordre de grandeur que la facture du réseau de données (divisé par dix).

-Ressources humaines : on peut constituer une seule équipe gérant les deux services (au lieu d'avoir deux équipes, chacune pour un réseau).

-Investissement : on peut utiliser un seul réseau physique et ainsi minimiser le coût en infrastructure.

Flexibilité

Côté services, cette intégration peut aussi permettre d'ajouter de nouvelles fonctions de communications aux équipements que nous utilisons :

- Un poste téléphonique va pouvoir communiquer avec n'importe quel ordinateur de l'Internet.
- Un ordinateur intégrera toutes les fonctions d'un téléphone .
- Le transport de la vidéo entre ordinateurs sera plus facilement généralisable
- L'intégration des messageries vocales et Internet sera très facile
- De nouveaux services de communications de groupe (« multicast téléphonique ») seront facilement réalisables.

Interopérabilité multi fournisseurs

Trop souvent par le passé les utilisateurs étaient prisonniers d'un choix technologique antérieur. La VoIP a maintenant prouvé que les choix et les évolutions deviennent moins dépendants de l'existant

1.2 Transmission de la voix sur IP

Dans ce paragraphe nous expliquons les différentes phases de transmission de la voix sur IP, ainsi que les différents protocoles utilisés à travers les différentes couches.

1.2.1 Les phases de transmission de la voix sur un réseau IP

La transmission de la voix sur IP et sur RTC sont deux opérations radicalement différentes [3]. Sur RTC la voix est transmise sous forme d'un flux constant, alors que sur un réseau IP cette opération passe par les étapes suivantes [5] :

Degitalisation de la voix (Voice digitalization)

Conversion de la voix de la forme analogique vers la forme numérique.

Annulation du bruit (Noise cancellation) :

Cette étape permet de séparer le signal de la voix de l'environnement du bruit.

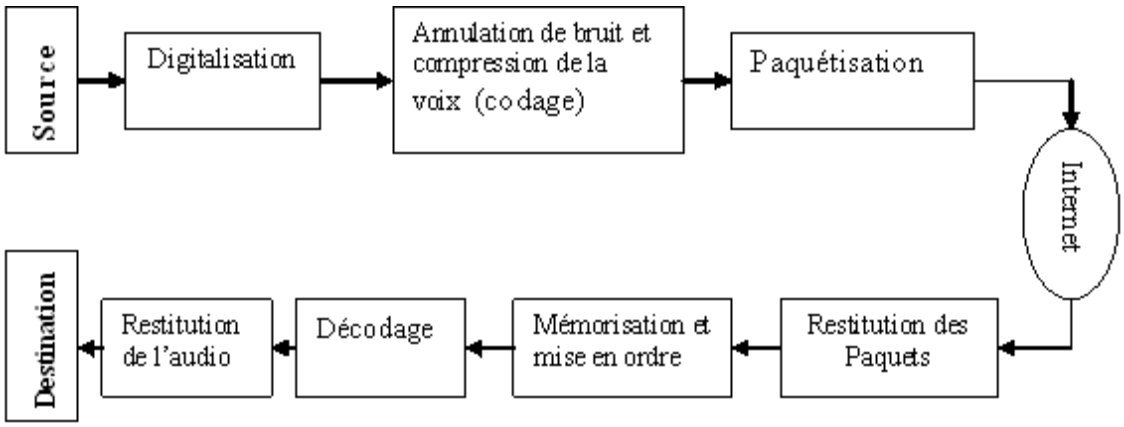

Figure 1.1 : Les différentes étapes de transmission de la voix

Compréssion de la voix (Voice compression)

La compression des paquets permet de réduire la largeur de la bande passante nécessaire pour la transmission de la voix.

Les deux dernières opérations (noise cancellation, voice compression) sont assurées par l'utilisation des codeurs/ décodeurs tel que : G.723, G.728, G.729, G.711....

Transmission

Après l'annulation du bruit et la compression, la voix est transmise sous forme des paquets IP en utilisant des protocoles de signalisation tel que H.323, SIP.

Restitution de l'audio

Cette étape permet la tranformation de la voix de la forme numérique vers la forme analogique.

1.2.2 La pile des protocoles Internet pour la transmission de la voix

La couche physique

La couche physique peut être un réseau local Ethernet, une ligne téléphonique (V.90) en mode point à point (PPP), ATM, réseau Ad Hoc (802.11).

La couche Internet

Le protocole Internet (IP) est utilisé dans cette couche pour router les paquets dans le réseau à leurs destinations.

La couche transport

Cette couche utilise deux octets pour le numéro du port afin de délivrer le datagramme à la destination. Quelques ports sont dédiés à des protocoles particuliers. Ces ports sont appelés numéro du port « bien connu », par exemple (HTTP utilise le port bien connu 80, tandis que SIP utilise les numéros du port bien connu compris entre 49152 et 65535). Trois protocoles sont utilisés dans la couche transport : TCP (Transmission Control Protocol), UDP (User Datagram Protocol) et TLS (Transmission Layer Sercurity).

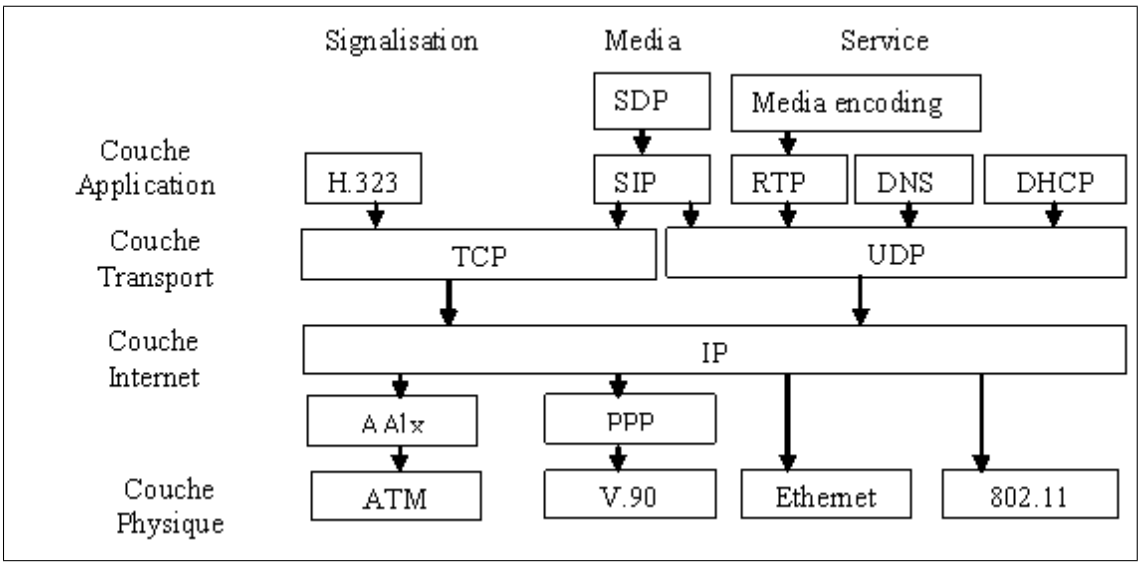

Figure 1.2 : La pile des protocoles Internet Multimedia

La couche Application (couche transport etendue)

Cette couche inclut les protocoles de signalisation comme SIP (Session Initiation Protocol), les protocoles de transport média comme RTP (Real time Transport Protocol). H.323 qui est un protocole de signalisation alternative au SIP, développé par l'union de télécommunication

internationale (ITU), protocole de description de sessions (SDP),HTTP, SMTP, FTP et Telnet.

La transmission de la voix sur IP nécessite des protocoles de transport des paquets tel que le protocole RTP/RTCP et des protocoles de signalisation pour l'établissement et la libération des sessions.

1.2.3 Les protocoles de transmission de la voix sur IP

Le protocole RTP/RTCP

IETF a adopté le protocole RTP (Real time Transport Protocol) avec son compagnon RTCP comme protocole de transport et de contrôle adapté aux applications ayant des propriétés temps réel. Le protocole RTP est employé au-dessus du protocole de datagramme comme UPD. Il fonctionne de bout en bout. Il ne permet pas la réservation des ressources et n'assure pas la retransmission automatique des paquets. Le protocole RTP offre à une application les possibilités pour [8] :

- Reconstituer la base de temps des flux de données audio, vidéo et temps réel en général.
- Détecter rapidement les pertes de paquets et informer la source dans des délais compatibles avec le service.
- Identifier le contenu des données pour en permettre la transmission sécurisée.

Constituants d'un paquet RTP

Les paquets de données RTP sont constitués d'une en-tête de 12 octets fixe (existe toujours) suivie de la charge utile [1]. La table1.1 donne les différents champs d'un paquet RTP.

V	P	X	CC	M	PT(7bits)	Sequence Number(16bits)
TimeStamp(32bits)						
SSRC:identifiant de la source de synchronisation						
CSRC:identifiant de la source de contribution						

Table 1.1: Format d'un en-tête RTP

Version (V): est constituée de 2 bits, ces deux bits indiquent la version du protocole, RTP.

Bourrage (padding): représenté par le champ P, quand ce dernier est égal à 1 cela indique que le paquet contient un ou plusieurs octets du bourrage.

Extension : représentée dans le paquet RTP par le champ X (voir table1.1).Lorsque le champ X égal à 1 alors une extension est ajoutée à l'en-tête RTP.

Le champ CSRC count : est constitué de 4 bits. Il indique le nombre des identifiant CSRC qui suivent l'en-tête fixe.

Le bit marqueur (Marker bit M):linterprétation du bit marqueur dépend du type de la charge utile. Il marque la fin de la trame pour la vidéo et le début de trame si les données transportées sont de type audio [1]

Type de la charge utile (Payload Type : PT): l'octet de type de la charge utile identifie le genre de la charge utile contenue dans le paquet (exemple : JPEG video).

Horodateur (TimeStamp): a une longueur de 32 bits. Il décrit l'instant de la génération des données contenues dans le paquet (la fréquence d'horodateur dépend du type de la charge utile).

Numéro de séquence (Sequence Number) : est codé sur 16bits, incrémenté par 1 à chaque envoi d'un paquet RTP. Il permet la détection des pertes et l'ordonnancement dans une série des paquets avec le même horodateur

Source de synchronisation (synchronisation source 32bits) : identifie d'une manière unique la source pendant une session RTP, sa valeur est générée d'une manière aléatoire et déterminée par un algorithme. Si un participant génère plusieurs flux dans la même session RTP chacun doit être identifié par un SSRC unique [2].

Le protocole RTCP

Le protocole RTCP est basé sur la transmission périodique des messages de contrôle à l'ensemble des participants de la même session. Il ne transporte que les informations de contrôle. Un paquet RTCP est constitué d'une entête fixe similaire à l'en-tête de RTP suivie par d'autres éléments qui dépendent du paquet RTCP transporté. On distingue cinq types de paquets RTCP indiqués dans la table 1.2.

SR (Sender Report)	Contient l'ensemble de statistiques de transmission et de réception en provenance des participants qui sont des émetteurs actifs
RR (Receiver Report)	Contient l'ensemble de statistiques en provenance de participants qui ne sont que des récepteurs et pas des émetteurs actifs.
SDES (Source Description)	Les paquets de description de source comprennent plusieurs éléments, dont le CNAME (canonical name).
BYE	Indique la fin d'une session
APP	Fonction spécifique à une application

Table 1.2: Les différents types de rapport RTCP

Fonctionnement du protocole RTP/RTCP

Le protocole RTP et son compagnon RTCP fonctionnent de bout en bout. A la réception des paquets SR (Sender Report) Les différents destinataires des paquets RTP analysent ces paquets et effectuent des statistiques sur les paquets perdus (utilisation de numéro de séquence pour calculer les paquets perdus), le délai de bout en bout (utilisation de l'horodateur), la gigue.

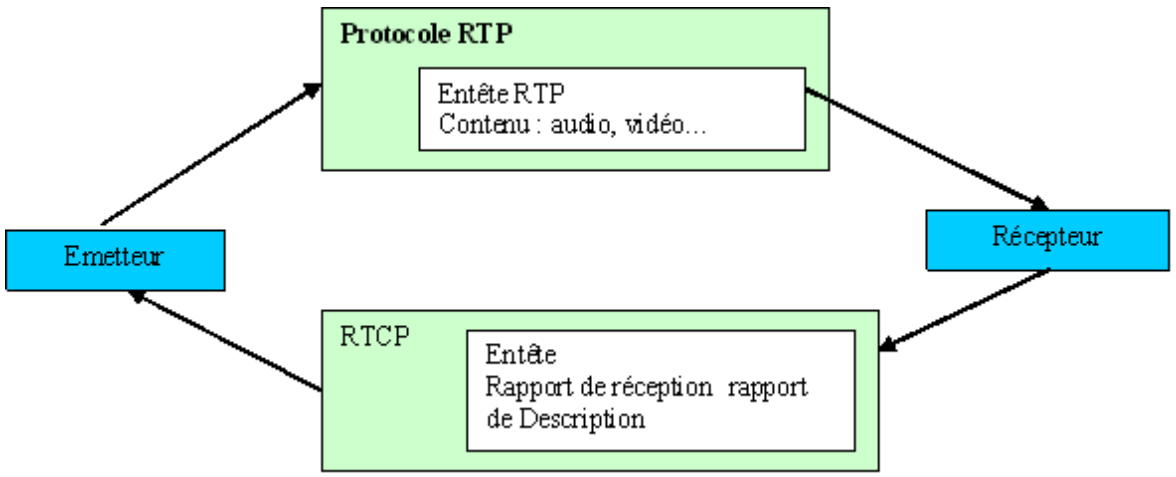

Figure 1.3 : Relation entre RTP et RTCP

Ensuite chaque récepteur encapsule toutes ces informations dans un rapport de contrôle RTCP et envoie ce dernier vers la source (l'émetteur). A la réception de ce rapport par l'émetteur ce dernier peut modifier ses paramètres d'émission (la largeur de la bande passante, nombre des paquets envoyés par session...) en fonction de ce feedback[1] reçu. La figure1.3 explique la relation entre RTP et RTCP.

1.2.4 Les protocoles de signalisation

La transmission de la voix sur IP est gérée par des protocoles de signalisation pour l'établissement de l'appel et la libération du circuit téléphonique. Trois protocoles sont actuellement définis : le protocole H.323, le protocole SIP et le protocole MGGP [2]. Nous présentons ici deux et nous orientons le lecteur intéressé par le troisième protocole (MGGP) à lire [2].

Le protocole H.323

Le protocole H.323 est une recommandation de ITU-T normalisé en 1996 pour les communications multimédias. Il appartient à la famille H.32x qui permet le support de

[1] feedback : les rapports envoyés par les récepteurs vers l'émetteur et qui contiennent l'ensemble des statistiques sur la QoS (les pertes, la gigue et le délai).

visioconférence sur différents réseaux. Il contient H.320 lié au réseau ISDN (Integrated service data network) et le H.324 lié au réseau PSTN (Public Switched Telephone Network). Ce protocole est composé des éléments suivants :

Un terminal utilisateur

Il s'agit soit d'un PC multimédia équipé d'un microphone et de haut parleur ou d'un téléphone IP doté d'une interface Ethernet pour se connecter au réseau (LAN).

La passerelle vers des réseaux classiques (Gateway)

Généralement ce sont des éléments matériels qui assurent le dialogue avec les systèmes téléphoniques traditionnels.

Le portier (Gatekeeper)

Est un élément logiciel. Il réalise deux fonctions essentielles:

- Traduction de l'adresse IP à une adresse téléphonique.
- La gestion des autorisations des appels.

Le contrôleur multipoint (Multipoint Control Unit)

Le MCU assure des fonctions de commande pour la mise en œuvre des conférences entre au moins trois points [2].

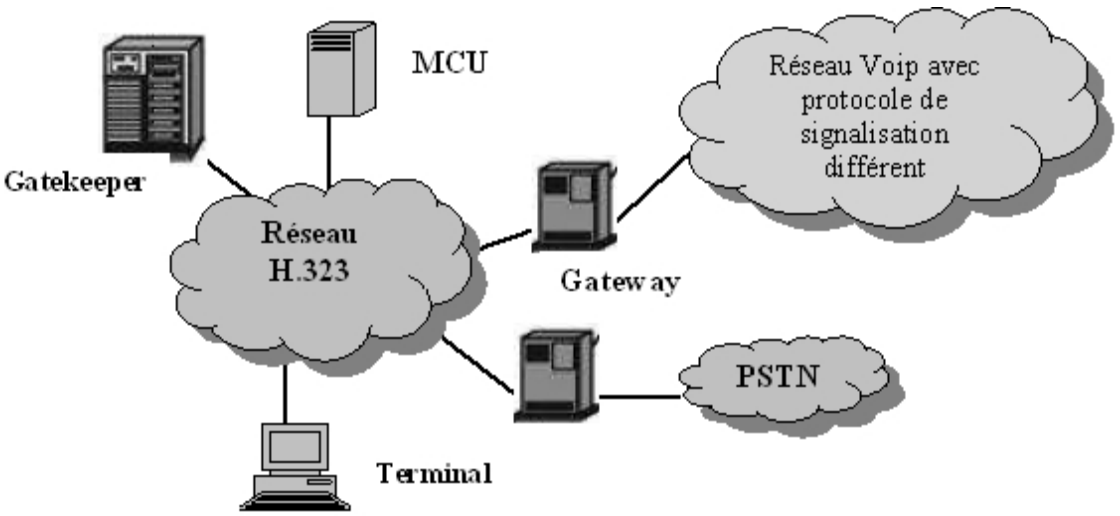

Figure 1.4 : Architecture H.323

Le protocole SIP

Le protocole SIP est un protocole de signalisation utilisé pour établir, modifier et terminer les appels vocaux et les sessions multimédias (multiparité) sur les réseaux IP (Intranet et/ou Internet). Dans le protocole SIP, un nœud peut se comporter comme un client et serveur. Pour établir un appel, un client envoie une demande SIP à un serveur, le serveur écoute la demande puis suggère une réponse. Les usagers SIP ont des adresses URL SIP semblables à des adresses électroniques (similaires à ceux de HTTP). Ces URL peuvent indiquer un usager dans un domaine (sip:usager@domaine) ou chez un hôte donné (sip:usager@hôte), à une adresse IP d'un hôte spécifique (sip:usager@addresse_IP), ou même à un numéro de téléphone (numéro E.164) auquel on accède par une passerelle IP/RTPC (sip:numéro_téléphone@passerelle).Il existe trois types de serveurs SIP [8]:

Les serveurs enregistreurs

Reçoivent et enregistrent des informations sur la localisation actuelle des clients ce qui aident à les localiser ensuite.

Les serveurs mandataires

Permetent d'acheminer les demandes d'un client vers la destination.

Les serveurs de re-acheminement

Redirigent les utilisateurs pour essayer un autre serveur SIP pour la prochaine fois.

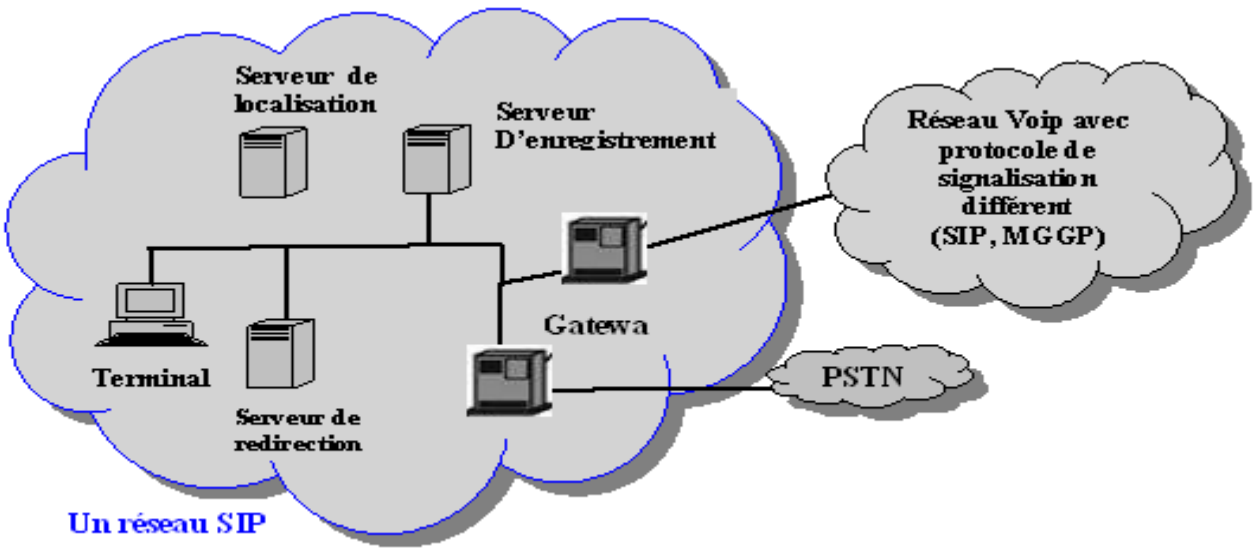

Figure 1.5 : Architecture SIP

Avec SIP les utilisateurs peuvent communiquer en mode point à point (communication entre deux machines ou en mode diffusif (plusieurs utilisateurs en multicast[1] via une unité de contrôle) ou le mode combinatoire (plusieurs utilisateurs pleinement interconnectés en multicast via un réseau à maillage complet). Pour établir et terminer des communications multimédias, SIP utilise les cinq fonctions suivantes :

- **User location** : Permet de localiser le poste terminal utilisé pour communiquer.
- **User capabilites** : Détermine quels médias vont être échangés (voix, vidéo, données,...), ainsi que les paramètres associés.
- **User availability** : Détermine si le poste appelé souhaite communiquer et autorise l'appelant à le contacter.
- **Call setup:** Avertit les parties appelée et appelante de la demande d'ouverture de session et met en place des paramètres d'appel.
- **Call handling** : Gère le transfert et la fermeture des appels.

1.3 Les paramètres affectant la transmission de la voix sur IP

1.3.1 Qualité du codage

La fonction primaire des codeurs/décodeurs est la conversion d'un signal de la forme vocale vers la forme numérique. La première opération du codage est l'échantillonnage[2] du signal analogique à une certaine fréquence d'échantillonnage et une certaine précision. La précision est caractérisée par le nombre de bits utilisés pour coder l'amplitude de chaque échantillon analogique [8]. Le choix de la fréquence et du nombre de bits utilisés représente un compromis débit/qualité du signal codé.

Le théorème d'échantillonnage établit qu'un signal analogique peut-être reconstruit à partir des échantillons numérisés si la fréquence d'échantillonnage est égale à au moins deux fois la largeur de la bande passante du signal original. L'oreille humaine est capable de percevoir une gamme de fréquences de 20 Hz à 20 kHz environ, correspondant à une bande passante de 20 kHz [8].Généralement plus le taux de compression est élevé par rapport à la référence 64kbits/s utilisée dans les réseaux téléphoniques commutés (RTC) moins la qualité de la voix est bonne [2].

Plusieurs algorithmes de compression de la voix ont été développés, mais les plus utilisés sont : le GSM 6.10, G.711, G.729, G.723, G.728. La table1.3 résume les caractéristiques principales de ces codeurs/décodeurs

[1] On entend par multicast le fait de communiquer simultanément avec un groupe d'ordinateurs identifié par une adresse spécifique (adresse de groupe).

[2] l'echantillonnage consiste à transformer un signal analogique (continu) à un signal numérique (discret), en capturant des valeurs à intervalle de temps régulier. C'est une etape nécessaire pour pouvoir enregistrer, analyser et traiter un signal par un ordinateur.

Codeur	G.711	G.723.1	G.726-32	G.729
Debit binaire(kb/s)	64	5.3/6.3	32	8
Charge utile(byte)	166	20/24	80	10
Retard du codeur/decodeur(ms)	0.125	90	0.3	30
Qualite de la parole	4.2	3.7-3.9	4.0	4.0

Table 1.3: Caractéristiques de quelques codeurs/decodeurs

Remarque

Qualité de la parole notée en MOS: la note moyenne d'opinion MOS est établie de manière normalisée selon cinq catégories: 1 = Mauvaise, 2 = Médiocre, 3 = Moyenne assez bonne, 4 = Bonne, 5 = Excellente [8].

1.3.2 Délai de bout en bout

Le temps nécessaire pour qu'un paquet traverse le réseau d'un point d'entrée à un point de sortie [2]. Il dépend des trois paramètres suivants:

- Type du media de transmission (temps de propagation) par exemple une liaison satellite est beaucoup plus lent qu'une liaison fibre optique.
- Le temps de traitement dans les équipements traversés dans le réseau.
- Taille des paquets (temps de sérialisation) : mesuré depuis l'émission du premier bit par la source jusqu'à la réception du dernier bit du même paquet au niveau du récepteur.

Ce paramètre est très important en particulier pour les applications intéractives comme la téléphonie IP qui exige un délai de bout en bout inférieur ou égal à 150ms.La table 1.4 résume les différentes valeurs recommandées par IUT-T.

Delais(ms)	Tolerance
moins de 150	Bonne interactivité
150-400	Les usagers peuvent noter quelques pertes dans l'interactivité
plus400	perte de l'interactivite

Table 1.4: Les délais recommandés par ITU-T pour la voix[4]

1.3.3 La gigue :

La gigue est définie par le temps inter arrivé entre les paquets. Elle dépend de :

- Type et volume du trafic sur le réseau.
- Type et nombre des équipements sur le réseau.

En générale dans les applications voix, la gigue doit être inférieure ou égal à 20ms. La figure1.6 illustre la distribution du delai et la gigue.

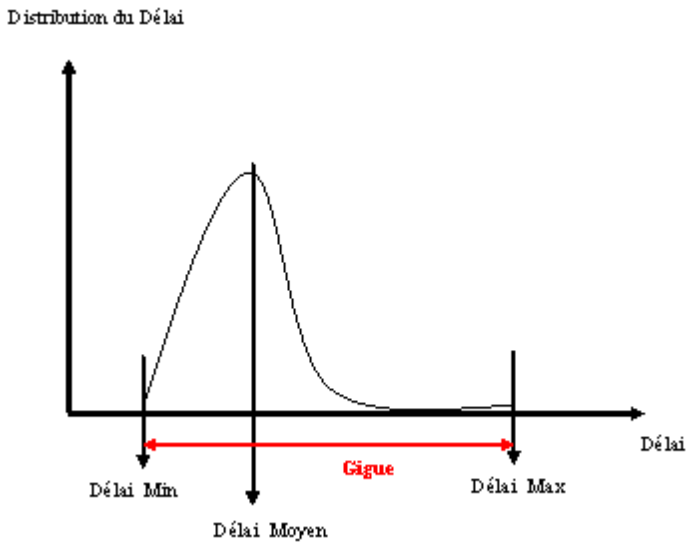

Figure 1.6 : Distribution de délai et illustration de la gigue

1.3.4 Perte des paquets

La gestion des pertes des paquets est devenue très difficile lorsqu'il s'agit des données multimédias interactives car il est impossible par exemple de ré-emettre un paquet voix perdu.

1.3.5 La largeur de la bande passante

Dans les réseaux, la bande passante est partagée entre plusieurs nœuds donc la largeur de la bande disponible est diminuée avec l'augmentation des nœuds accédant à la bande.La figure 1.7 résume les différents problèmes affectant la transmission de la voix.

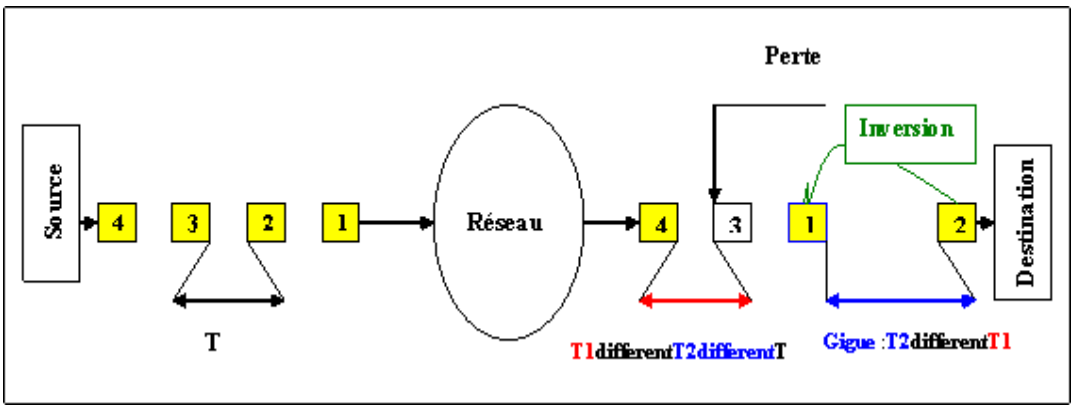

Figure 1.7 : Les principaux problèmes affectant la transmission de la voix sur IP

1.4 Solutions pour améliorer la qualité de service

Le problème de la qualité de service dans les réseaux constitue un vaste sujet de recherche et a fait l'objet de nombreuses techniques proposées dans la littérature. En dehors des techniques tendant à organiser l'architecture de communication de façon à réduire les délais d'acheminement des messages tout en garantissant une bonne disponibilité du support de transmission (en travaillant notamment sur le déploiement du réseau en utilisant des algorithmes de placement appropriés ou la prise en compte de la QoS au niveau de la couche liaison), les autres peuvent être classées dans les catégories suivantes : Le contrôle de congestion, le contrôle de débit, la réservation de ressources, la différenciation de services et l'intégration de la QoS dans les décisions de routage des paquets.

1.4.1 Le contrôle de congestion

Lorsqu'un paquet arrive dans un routeur, il est placé dans une file d'attente. Si le débit des paquets en entrée d'un routeur est plus grand que le débit de sortie, le nombre de paquets dans la file d'attente augmente. La taille des files étant limitée, une congestion survient lorsque la file est pleine. Les paquets devant être placés dans la file pleine sont alors détruits. Il est donc essentiel de limiter d'une part, l'apparition de congestion et d'autre part, leur durée. De nombreux travaux se sont orientés dans ce sens, parmi ces travaux on trouve les suivants:

1.4.1.1Le démarrage lent (Slow Start)

Dans sa version originale, le protocole TCP permet d'envoyer dès le début de la transmission, un nombre de paquets correspondant à la taille de sa fenêtre d'émission. La taille de cette fenêtre est déterminée à l'issue d'une négociation entre la source et la destination. En 1988 Van Jakobson a montré que cela peut être la cause de congestion dans le réseau. Ensuite, il a proposé d'utiliser une fenêtre supplémentaire, appelée : la fenêtre de congestion. Lorsqu'une nouvelle connexion est établie, la taille de cette fenêtre (cwnd) est initialisée à un segment. Chaque fois qu'un acquittement est reçu par la source, la taille de la fenêtre de congestion est incrémentée d'un segment. Le nombre de segments transmis (taille de la fenêtre d'émission), est borné par le minimum de la taille de la fenêtre de réception de la destination et de la fenêtre de congestion. La source commence par envoyer un segment, et attend son acquittement. Lorsque l'acquittement est reçu, la fenêtre de congestion est incrémentée d'un segment. La source émet donc deux segments. Pour chacun des deux acquittements reçus pour ces segments, la fenêtre de congestion est à nouveau incrémentée d'un segment, c'est-à-dire passe à 4 segments, et ainsi de suite, la taille de la fenêtre de congestion augmente ainsi exponentiellement. Lorsqu'un paquet est perdu, cela signifie que la fenêtre de congestion est devenue trop grande, on peut alors faire appel à la technique de l'évitement de congestion.

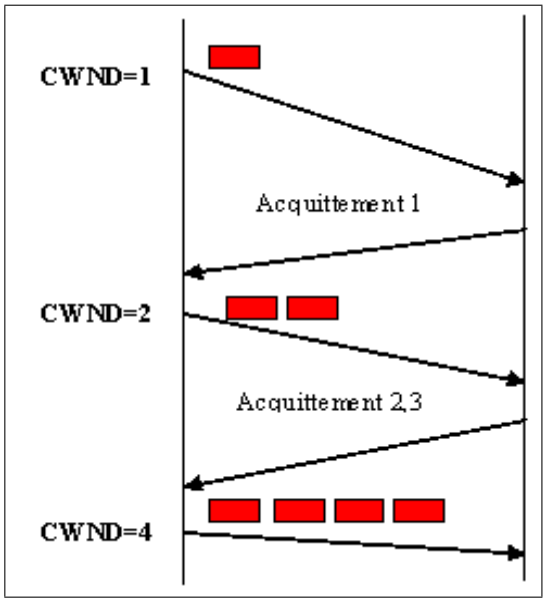

Figure 1.8 : Slow Start

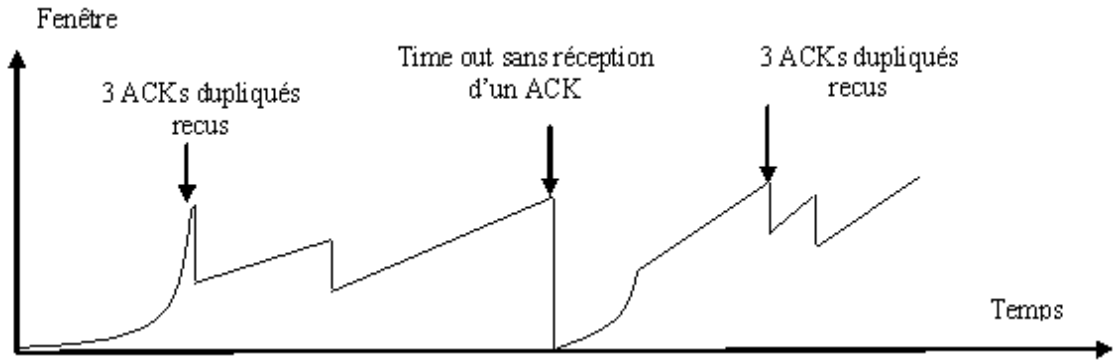

Figure 1.9 : Influence de slow start sur le délai

1.4.1.2 L'évitement de congestion (Congestion Avoidance)

La source détecte qu'un paquet a été perdu lorsqu'elle reçoit des acquittements dupliqués ou bien si elle ne reçoit pas l'acquittement d'un paquet après une période donnée (time out). Le nombre de paquets corrompus à cause d'un défaut de liaison est très faible dans les réseaux filaires. On peut considérer que la perte d'un paquet est le signe d'une congestion dans le réseau. Dans ce cas, l'évitement de congestion permet de réduire le débit de la source, en réduisant la taille de sa fenêtre. Pour cela, on applique un seuil au démarrage lent (Slow Start Threshold, ssthresh).lorsqu'une perte de paquet est détectée, on affecte à ssthresh une valeur égale à la moitié de la taille de la fenêtre d'émission, et on réduit la taille de la fenêtre de congestion (cwnd) comme suit :

- Si la perte de paquet a été détectée grâce à des acquittements dupliqués, cwnd=ssthresh.
- Si la perte de paquet a été détectée après un time-out, cwnd=1.

Ensuite, l'augmentation de la fenêtre de congestion peut être de deux natures :

- Si cwnd<ssthresh, on applique le démarrage lent. Par conséquent, la taille de la fenêtre de congestion augmente exponentiellement.
- Si cwnd>ssthresh, on applique l'évitement de congestion. A la réception de chaque acquittement, on augmente cwnd d'une taille fixe. La taille de la fenêtre de congestion augmente donc linéairement.

Il est à noter que le démarrage lent et l'évitement de congestion sont actuellement implémentés dans le réseau Internet, dans le protocole TCP. Cependant, ces deux techniques ne permettent la réduction du débit des sources que lorsqu'une congestion a débuté. L'apparition de congestion n'est donc pas écartée. C'est pourquoi des algorithmes permettant la prévention des congestions, tels que RED et WRED, ont été proposés.

1.4.1.3 RED (Random Early Detection) et WRED (Weighted Random Early Detection)

Proposé en 1993 par Sally Floyd et Van Jakobson, le mécanisme RED est une technique préventive permettant d'éviter les congestions. Son principe consiste à détruire arbitrairement des paquets dans les files d'attente, lorsque celles-ci sont remplies au-delà d'un certain seuil. Plus la file se remplit, plus le nombre de flux concernés par la destruction arbitraire de paquets est grand. Grâce au mécanisme d'évitement de congestion, les flux transportés par les paquets détruits voient leur débit réduit.

RED est équitable, puisque les flux qui subissent des destructions de paquets sont choisis arbitrairement. Au contraire, pour privilégier certains flux par rapport à d'autres, on peut appliquer différentes probabilités de destruction. C'est le but des techniques Weighted RED (WRED) et Enhanced RED, où les paquets des flux privilégiés, donc plus prioritaires, sont moins exposés à la destruction arbitraire. WRED distingue la priorité des paquets grâce au champ de priorité présent dans chaque en-tête de paquet.

Les techniques de régulation des sources que nous avons présenté ci-dessus s'appliquent aux flux individuellement. Une autre démarche consiste à imposer plus généralement des limitations de débit à chaque routeur.

1.4.2 Le contrôle de débit

Le contrôle de débit (traffic shaping), est une approche globale. Son principe est de limiter le débit des routeurs d'accès (les routeurs d'extrémités), et donc le nombre de paquets injectés dans le réseau, afin de ne pas dépasser un certain seuil en terme d'utilisation des ressources disponibles. Tous les flux ne sont pas sujets à l'évitement de congestion. En effet, si le débit des flux TCP est sensible à la charge du réseau, car chaque paquet est acquitté, ce n'est pas le cas pour les flux UDP qui sont émis avec un débit qui peut être très grand, quel que soit l'état du réseau. L'aspect global du contrôle de débit permet, en limitant le

débit total des routeurs, d'imposer des limites à ces flux indésirables. Le Leacky Bucket est l'une des techniques permettant le contrôle de débit.

1.4.2.1 Seau troué

Le Seau troué ou «Le Leacky Bucket », permet de réguler le trafic émis par les routeurs dans le réseau. Ainsi les paquets devant être émis, sont d'abord placés dans une file d'attente puis seront envoyés à des intervalles réguliers. Cette méthode permet donc d'éviter les émissions en rafale et donc diminue le débit du réseau.

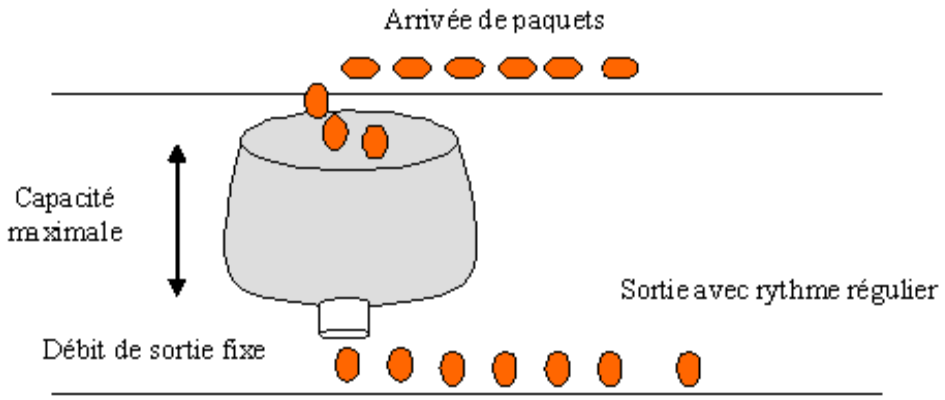

Figure 1.10 : Seau troué

La taille de la file d'attente et le débit d'émission sont configurables. Toutefois, cette méthode ne permet pas de s'adapter à la charge du réseau : le débit d'émission reste constant quelle que soit la charge en plus la taille du réseau est limitée donc le nombre des paquets à détruire augmente. Une technique dérivée permet au contraire de tenir compte des ressources disponibles dans le réseau : le seau à jetons (le Token Bucket).

1.4.2.2 Seau à jetons

Pour une meilleure gestion des ressources, il est préférable de moduler le débit des routeurs en fonction de la charge du réseau. Le seau à jeton ou "Le Token Bucket", utilise la notion de jeton pour évaluer la charge du réseau.

Le nombre de jetons disponibles est inversement proportionnel à la charge du réseau. Le débit du routeur est conditionné par le nombre de jetons disponibles. Ainsi, le Token Bucket permet l'émission en rafale lorsque le réseau est peu chargé.

Un autre moyen d'améliorer la qualité de service dans l'Internet est la réservation des ressources nécessaires à la transmission des informations.

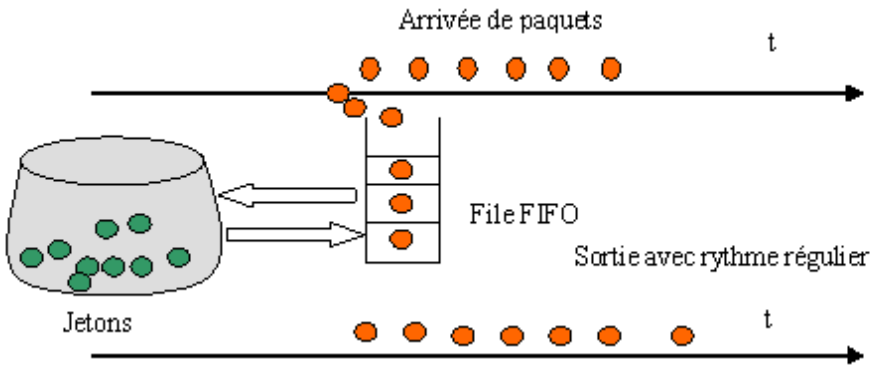

Figure 1.11 : Seau à jetons

1.4.3 Reservation des ressources

IntServ

L'architecture IntServ est développée par le groupe IntServ (Integrated Services). Elle offre une QdS par flux. le groupe IntServ a défini deux autres types de service en plus du service best effort : le Controlled Load (CL) et le Guaranteed Service (GS) [10]. Le CL : permet de contrôler la charge dans le réseau.Il propose un service de bout en bout exprimable de façon qualitative en terme de bande passante, il assure que la transmission se fera comme sur un réseau peu chargé (pas de congestion). Le GS propose un service exprimable de façon quantitative en terme de bande passante et de délai de transit maximal c'est-à-dire il garantit que tous les paquets d'un même flux arriveront en un temps borné défini par l'application qui utilise le service. Le modèle IntServ repose sur les trois points fondamentaux suivants :

- La réservation des ressources est effectuée par session au niveau de chacun des hôtes et de routeurs du chemin de données.
- Cette réservation est effectuée en utilisant le protocole RSVP (Ressource Réservation Setup Protocol) (pour plus de détaille sur RSVP, nous renvoyons le lecteur à [8]). L'émetteur initialise une requête de réservation et les récepteurs effectuent cette requête. La requête est propagée d'un routeur à l'autre. Chaque routeur effectue un contrôle d'admission en tenant compte des ressources disponibles localement et les caractéristiques du trafic fournies avec la réservation et enregistre en cas de succès un état de réservation pour la session considérée [9].
- Le maintien des réservations est assuré par la réception de messages de rafraîchissement émis par les émetteurs.

Chaque routeur dans l'architecture IntServ dispose des fonctionnalités suivantes:

- Le protocole de réservation de ressource: qui signale le chemin à établir en sollicitant des réservations de bande passante sur chaque routeur traversé.

- Le contrôle d'admission : permet d'autoriser l'arrivée d'un nouveau flux muni de sa QoS sans perturber la QoS des flux existants.
- Classifieur: classe les paquets de flux admis dans des classes spécifiques.
- L'ordonnanceur: de paquets détermine l'ordre de service des paquets.

Deux principaux problèmes sont posés par IntServ [9]

- La résistance au facteur d'échelle : il s'agit ici de définir comment minimiser le nombre d'états de réservation qu'un routeur est susceptible de maintenir et le nombre de messages de signalisation nécessaires au maintien de ces états par rafraîchissement périodique.
- Comment définir une architecture de bout en bout permettant d'interfacer les applications avec les services IntServ.

RSVP

L'IETF, a conçu le protocole RSVP pour la réservation de ressources au sein d'un réseau Internet. Il peut être utilisé pour assurer la qualité de service et gérer les ressources de transport des données sur le réseau pour les sessions point à point (unicast) et point à multipoints (multicast).RSVP est un système de contrôle et de signalisation qui donne la possibilité de réserver la bande passante nécessaire pour un bon fonctionnement d'une application. Il s'agit d'un besoin qui touche principalement les flux multimédias, plus sensibles aux aléas de l'acheminement que les flux de données pures de fait de leurs contraintes temporelles.

- RSVP est basé sur le concept de session. Une session est composée d'au moins un flux de données et le triplet : adresse de la destination, port de la destination, identification du protocole.
- Dans RSVP, un flux est défini comme étant n'importe quel sous-ensemble de paquets d'une session, en d'autres termes, un flux est un sous-ensemble des paquets envoyés à une destination par une même source.
- Des messages PATH sont émis périodiquement vers la destination. Ces messages décrivent le flux de données émis par chaque source (en termes de qualité de service) et établissent dans les routeurs un « état de chemin » (PATH state) par flux.
- Des messages RESV sont émis périodiquement vers les émetteurs. Ces messages décrivent les réservations à effectuer. Grâce aux états de chemins établis par les messages PATH, les messages RESV suivent le chemin inverse des paquets de données. Ainsi, les réservations sont effectuées dans les routeurs relayant ces données. Comme les ressources sont réservées à la requête des receveurs, RSVP, offre donc un style de réservation dite "orientée récepteur ".
- Pour améliorer le temps de réponse de RSVP aux changements dynamiques du routage dans le réseau, le mécanisme dit "réparation locale " a été introduit. Lorsqu'un nœud RSVP détecte un changement de route, il envoie immédiatement un message PATH

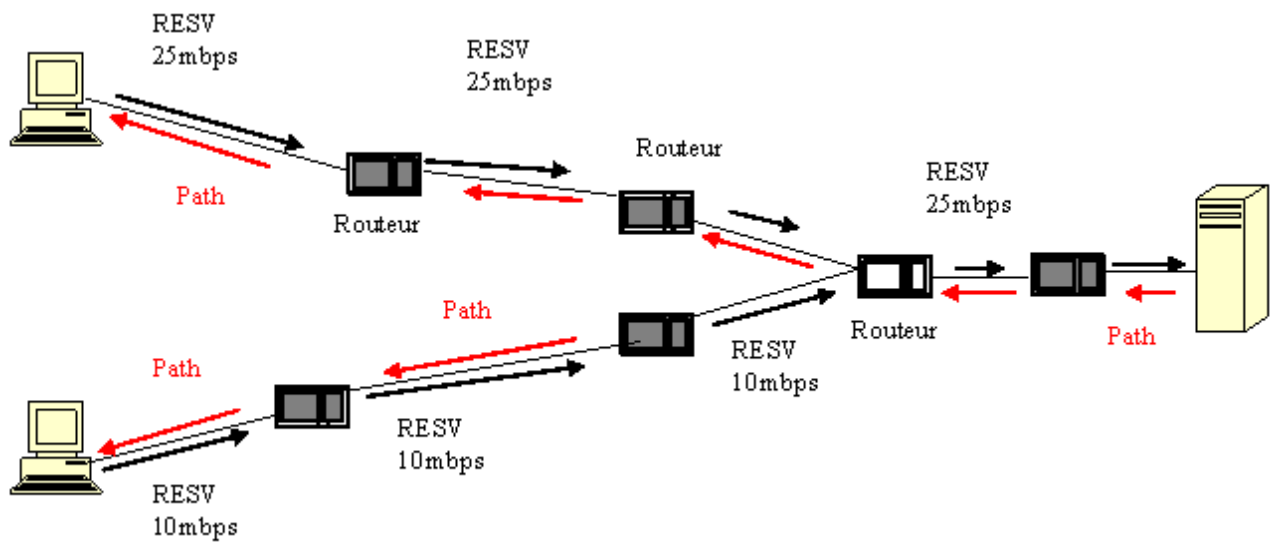

Figure 1.12 : Fonctionnement du protocole RSVP

par flux re-routé le long de la nouvelle portion de route. Par ailleurs, lorsqu'un nœud RSVP reçoit un message PATH pour un flux, mais pour lequel l'état de chemin qu'il a mémorisé diffère de celui indiqué par ce message, le nœud met à jour son état de chemin. Il transmet immédiatement, le long du nouveau tronçon de route, un message RESV décrivant la réservation effectuée pour ce flux.

- Un temps de vie (timer) est associé à chaque ressource réservée. La valeur de ce temps de vie est réinitialisée à chaque fois qu'un message RESV confirme l'utilisation de cette ressource. Si le temps de vie vient à s'écouler, la ressource correspondante est libérée.
- Des messages de relâchement (teardown messages) peuvent être utilisés pour relâcher les états de chemins et les réservations. Les requêtes de relâchement sont initiées soit par l'émetteur, le récepteur ou par n'importe quel nœud RSVP intermédiaire (à l'expiration d'un temps de vie).

Inconvénients de RSVP

RSVP maintient l'état des ressources d'un flux. Lorsque le nombre des utilisateurs augmente, le nombre des états augmente. Par conséquent, le trafic généré pour le rafraîchissement devient plus important ce qui diminue les performances du système

1.4.4 Differentiation de service

DiffServ

L'architecture DiffServ (différentiation de service) est normalisée par le groupe DiffServ de l'IETF [8].contrairement au service best effort qui fait son mieux pour transporter les paquets, mais sans les différencier les uns des autres, DiffServ fournit une QoS différenciée aux paquets traversant un réseau tout en repoussant la complexité des traitements vers l'extrémité afin de ne pas surcharger le cœur du réseau [10].A chaque type de flux est associé une priorité. Ainsi, la priorité la plus élevée (trafic premium) correspond aux flux requérant la plus grande qualité de service (flux temps réel), et la priorité la plus faible, aux flux ne nécessitant pas des contraintes particulières. Les priorités intermédiaires se déclinent suivant les niveaux de qualité de service requis. La classification est effectuée au niveau des paquets en considérant le champ TOS (Type Of Service d'IPv4 que l'IETF a redéfini en champ DS (DiffServ)) de leur en-tête. Ce dernier inclus la priorité et l'identificateur du flux. Les paquets sont ensuite placés dans l'une des files d'attente existantes dans chaque interface de sortie, chacune d'entre elles correspondant à un service différent. Enfin, c'est la politique d'ordonnancement de ces files d'attente qui détermine l'ordre dans lequel les paquets sont émis vers le routeur suivant. La différenciation de services permet dans une situation d'encombrement d'éliminer certains paquets non prioritaires pour en protéger d'autres. Les fonctionnalités de l'architecture DiffServ sont présentées dans la figure 1.13.

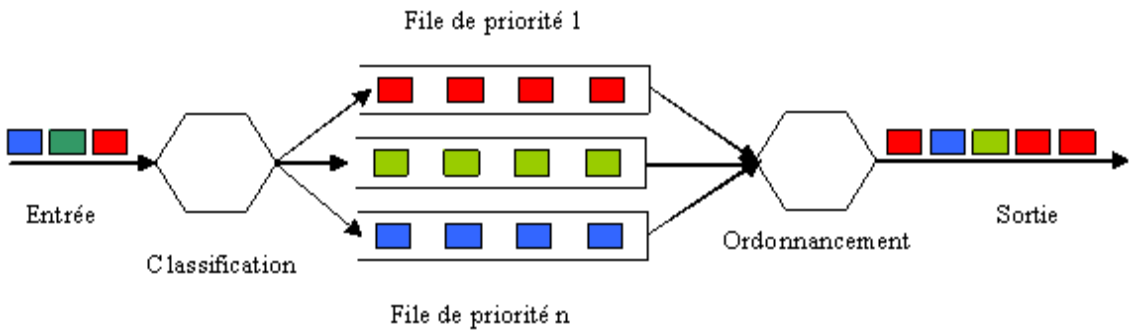

Figure 1.13 : Classification et ordonnancement des paquets dans DiffServ

A l'entrée du paquet dans le réseau

Une priorité est affectée à chaque paquet. Cette priorité peut dépendre de l'identité de l'émetteur (par exemple un utilisateur ayant souscrit un abonnement préférentiel) ou de la nature du trafic. Dans ce dernier cas, la priorité dépend des contraintes de qualité de service requises.

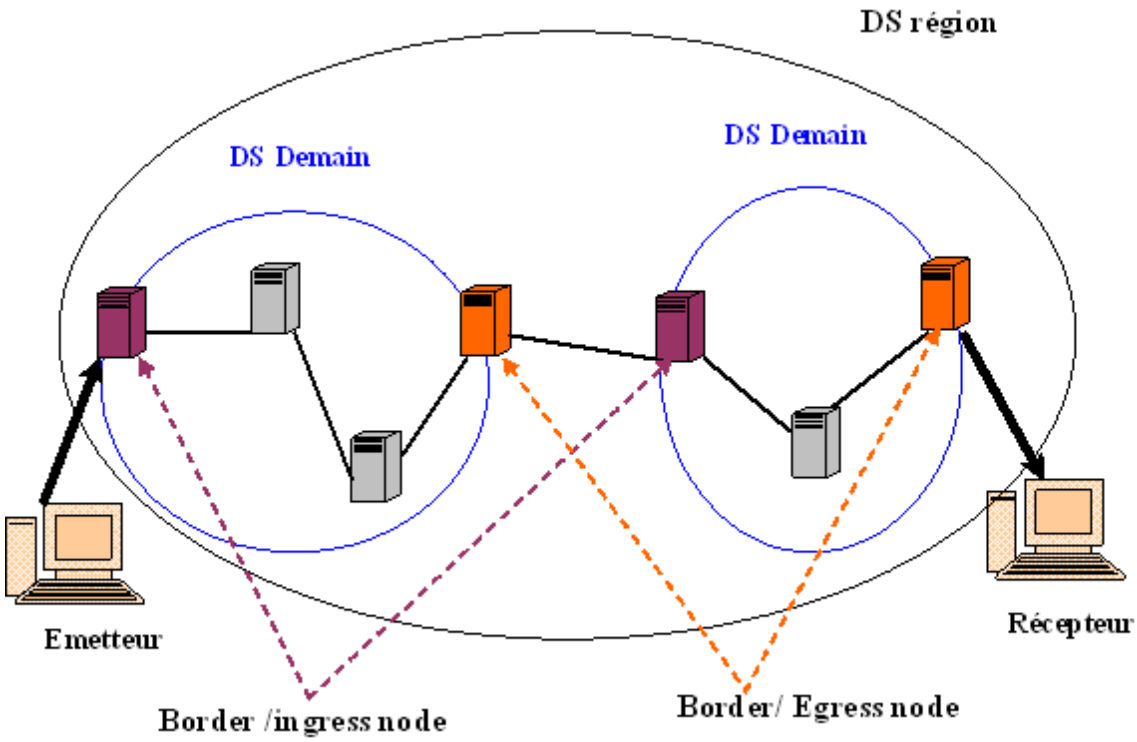

Figure 1.14 : Architecture DiffServ

Au niveau des routeurs

DiffServ distingue deux types de routeurs :

Les routeurs d'extrémité : se chargent de la mise en forme et de la classification du trafic. Une de leurs fonctions est d'attribuer une étiquette DSCP à tous les paquets entrant dans le domaine.

Les routeurs du cœur : ces routeurs traitent les paquets selon l'étiquette DSCP.L 'IETF a défini deux services DiffServ en plus de service best effort (c'est-à-dire deux PHP : Per hop behaviour) [8].

Le PHP de transmission express (EF : Expedited Forwarding) : les flux marqués par EF reçoivent un service de transmission qualitativement supérieur que les paquets best effort, le service EF rencontre une file d'attente qui devrait être courte et traitée rapidement afin d'avoir un délai et taux de perte faible.

Le PHP de transmission assurée (AF : Assured forwarding) : le PHB AF est destiné à des services plus généraux. Cette famille de PHB est scindée en 4 classes garantissant de fournir une bande passante et un délai minimum, chaque classe comprenant 3 niveaux de priorité de rejet qui définissent l'importance relative d'un paquet dans une classe particulière en cas d'encombrement.

L'architecture DiffServ est représentée dans la figure1.14.

La classification des paquets dans les routeurs se fait suivant leur priorité et leur placement dans la file d'attente.On retrouve plusieurs techniques d'ordonnancement des files d'attente dans le modèle Diffserv.On cite les suivantes:

Premier arrivé premier servi (FIFO: First In First Out)

L'ordonnancement FIFO est la politique la plus simple. C'est celle qui est implémentée de façon standard dans le réseau Internet, ne permettant que le service Best Effort. Le principe FIFO (First In, First Out) garantit que les paquets sont transmis en sortie dans leur ordre d'arrivée. Toutefois, lorsque la file est pleine, les paquets entrants sont détruits. Par ailleurs, il est possible d'implémenter une différenciation de services avec une seule file, en imposant que les paquets moins prioritaires soient détruits avant que la file ne soit pleine, laissant ainsi leur place aux paquets plus prioritaires.

Files prioritaires

La méthode des files prioritaires utilise donc une file par classe de service. A chacune de ces files est associée une priorité stricte, c'est-à-dire que les paquets d'une file ne sont émis sur l'interface de sortie que lorsque les files plus prioritaires sont vides.

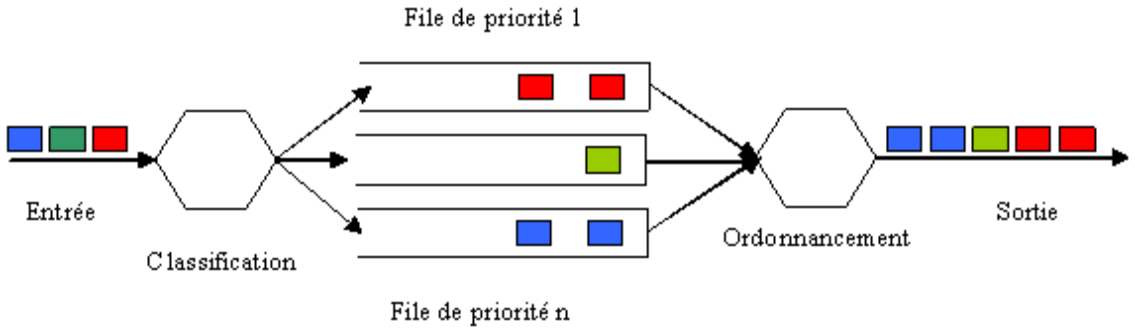

Figure 1.15 : Files prioritaires

La technique des files prioritaires présente toutefois un inconvénient majeur. En effet, un trafic prioritaire important engendre une famine des trafics moins prioritaires. Pour pallier ce défaut, on peut appliquer un contrôle de débit aux flux prioritaires ou bien imposer un débit minimal aux files de moindre priorité.

GPS (General Processor Sharing)

La technique GPS associe une priorité et une file d'attente à chaque flux. L'ordonnanceur retire ensuite de chaque file, à tour de rôle, une quantité de données proportionnelle à sa priorité et l'émet. Les politiques d'ordonnancement décrites ci-après ont été dérivées du GPS.

Round Robin(RR)

La technique du Round Robin est très simple. Les files d'attente associées à chaque flot sont examinées à tour de rôle. Un paquet est extrait de chaque file non vide, puis est émis sur l'interface de sortie . Chaque flot est donc traité équitablement.

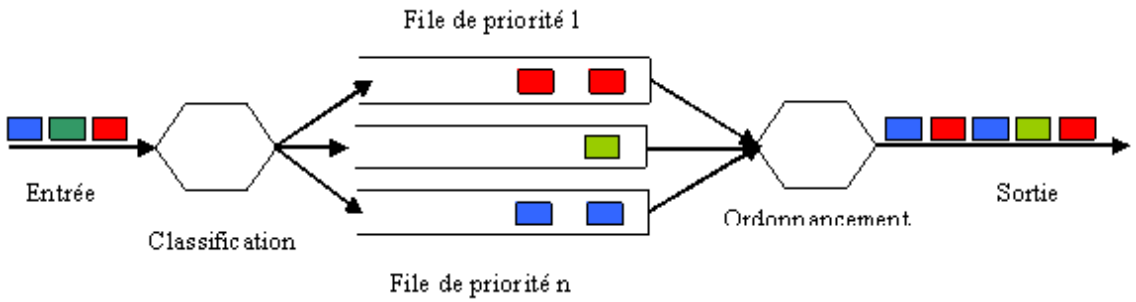

Figure 1.16 : Round robin

Weighted Round Robin(WRR)

La méthode du Weighted Round Robin fonctionne sur le modèle du Round Robin, mais ici, à chaque flux (et donc à chaque file), est associée une priorité. L'Ordonnanceur examine alors les files non vides, extrait un nombre de paquets proportionnel à la priorité, puis les émet. Certains flux sont donc privilégiés par rapport à d'autres.

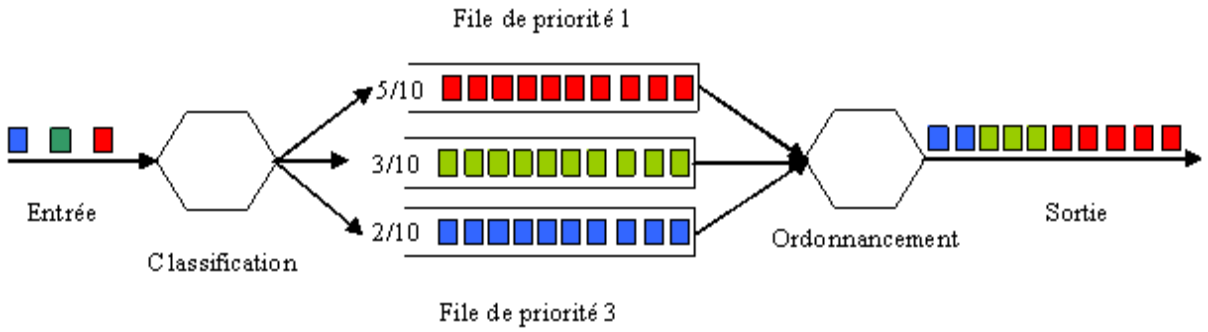

Figure 1.17 : Weighted Round Robin

Le principal inconvénient des techniques RR et WRR est qu'elles ignorent la taille des paquets. Un flux composé de paquets de plus grande taille est donc privilégié, en termes de quantité de données transmise par les routeurs. Les techniques DRR et WDRR ont donc été proposées pour y remédier. Le DRR (Deficit Round Robin) permet de pondérer la quantité de données extraites des files pour chaque flot par la taille des paquets composant ce flot grâce à la méthode des quantum . De même, WDRR fonctionne sur le même schéma, tout en tenant compte des priorités des flux.

FQ (Fair Queuing) et WFQ (Weighted Fair Queuing)

Toujours dans le but de tenir compte de la taille des paquets, la technique du Fair Queuing consiste à effectuer un Round Robin sur les files d'attente, mais en simulant l'extraction d'un seul bit de chaque file non vide. En fait, on calcule l'instant t où un paquet sera entièrement extrait avec un tel ordonnancement, et on fixe l'extraction effective du paquet à cet instant t.

La technique du Weighted Fair Queuing fonctionne sur le même principe, tout en assignant un poids à chaque file.

Les techniques basées sur le GPS apportent donc un certain nombre d'avantages

- Partage équitable des ressources.
- Différenciation des services.
- Elimination des phénomènes de famine.

Toutefois, ils utilisent une file par flux. Ce principe pose naturellement des problèmes d'échelle. En effet, plus le nombre de flux est grand, plus le nombre de files est grand, et plus les tâches de classification et d'ordonnancement sont coûteuses. Les propriétés remarquables de cette famille de techniques ont été conservées, tout en réduisant le nombre de files à gérer.

D'autres travaux ont été effectués pour améliorer la qualité de service. Parmi ces travaux on trouve les suivants:

Le protocole STCP (Stream Control Transmission Protocol)

Le protocole STCP, est un protocole de transport à fiabilité totale se déployant sur un service paquet de niveau réseau sans connexion, offert par exemple par le protocole IP. Il est unicast et orienté session, une session étant définie comme une association établie entre deux hôtes. Dans le cas où un hôte dispose de plusieurs adresses IP (à plusieurs interfaces), les adresses sont échangées lors de l'établissement de la session. Il est multi-homing (on appelle multihoming le fait que plusieurs adresses IP peuvent correspondre à une session). STCP inclut un mécanisme pour contrôler les erreurs. Ce mécanisme permet de détecter les pertes, les ruptures des séquences, la duplication ou la corruption des paquets. Un schéma de retransmission est utilisé pour corriger ces erreurs. SCTP utilise le principe de Selective ACKnowledgement : SACK pour la confirmation de la réception des données. Les retransmissions sont faites après expiration d'un timer ou sur interprétation du SACK. SCTP est orienté message (ce qui le rapproche par cet aspect du UDP). Chaque paquet contient un en-tête commun et une partie donnée (contenant soit des données utilisateurs soit des données de contrôle). En fait, bien qu'il soit orienté message, plusieurs données peuvent être contenues dans le même paquet, mais seront délivrées à l'application avec le format des messages initiaux. STCP offre un service de multiplexage/démultiplexage entre flux par exemple un flux peut-être découpé en plusieurs flux pouvant avoir chacun un chemin différent vers la destination. Les paramètres de contrôle de flux sont négociés à

l'établissement de la connexion. Le récepteur informe l'émetteur de sa taille de buffer et la taille de la fenêtre de congestion.

Le fait que SCTP offre un service totalement fiable entraîne une incompatibilité avec les applications multimédias ayant des contraintes en terme de débit, de délai ou de la gigue. Une autre version de STCP est proposée en 2003 qui permet à l'utilisateur de spécifier une durée de vie pour leur message.

Le protocole DCCP

Le protocole DCCP (Datagram Congestion Control Protocol), développé par Kohler en 2002, offre un service de transport non fiable pour des flux datagram (type UDP) mais intégrant un mécanisme de contrôle de congestion, ce qui permet aux applications utilisant habituellement UDP de ne pas avoir à implémenter des mecanismes de controle de congestion . Le but de DCCP est d'offrir l'efficacité pour certaines applications qui utilisent UDP tout en respectant les autres flux (TCP) du réseau. Les mécanismes de contrôle de congestion sont négociés pour les deux sens de la connexion entre les hôtes au moyen d'un identifiant appelé : Congestion Control Identifier (CCID).

DCCP peut être utilisé par toutes les applications présentant des contraintes temporelles et qui sont capables de s'adapter aux fluctuations de débit imposées par les mécanismes de contrôle de congestion.

1.5 Conclusion

La transmission de la voix sur IP ou la téléphonie Internet connue aussi sous le nom « la téléphonie sur IP » complémente et remplace peu à peu la téléphonie classique (RTC : réseau téléphonique commuté). Cette application a connu une croissance énorme ces dernières années par exemple U S A Forrester Research a prédit que le marché VoIP s'est développé de 30 millions de dollars en 1998 à un marché de 2 milliards de dollars en 2004.

D'autre part les réseaux sans fil et en particulier les réseaux Ad hoc ont pris un grand intérêt dans le monde de télécommunication à cause de leur faible coût et simplicité d'installation dans les différents endroits (désert + guère...). Mais assurer une bonne qualité de service pour la voix sur IP et en particulier sur les réseaux mobiles Ad hoc est une tache très difficile dû aux caractéristiques de ces réseaux à savoir : la mobilité, l'énergie faible, ...

Pour pallier à ces problèmes de QoS, plusieurs modèles ont été proposés dans la littérature soit pour les réseaux filaires ou les réseaux Ad hoc. Dans ce chapitre nous avons présenté des modèles de QoS pour les réseaux filaires et on a classifié en quatre catégories à savoir : le contrôle de congestion (démarrage lent, l'évitement de congestion...), le contrôle de débit (seau troue, seau à jeton), les méthodes de réservation des ressources (RSVP, IntServ) et des méthodes basées sur la différenciation de service telles que DiffServ. Ces protocoles permettent de fournir une bonne qualité de service dans les réseaux filaires, mais

ne permettent pas de fournir cette qualité pour les réseaux sans fil et en particulier les réseaux Ad hoc, car ces protocoles ne prennent pas en compte la qualité dynamique des nœuds de ces réseaux. Dans le prochain chapitre, nous présentons les réseaux Ad hoc et les caractéristiques de ces réseaux. Ensuite, on passe vers les modèles de QoS dans les réseaux Ad hoc.

2

Les réseaux sans fil et le routage

2.1 Introduction

Le monde des réseaux sans fil a connu un grand succès ces dernières années. Ce succès est dû essentiellement à la chute de coût des appareils sans fil ainsi qu'à la miniaturisation des composants électroniques. Le réseau sans fil n'a besoin d'aucune installation fixe ce qui rend son déploiement facile et rapide. La topologie d'un réseau sans fil est dynamique, elle change d'une manière imprévisible. Lorsque les nœuds n'arrivent pas à s'entendre directement, des nœuds intermédiaires peuvent jouer le rôle de routeur pour transmettre les paquets de données de la source vers la destination.

Dans ce chapitre, nous commençons par la présentation des deux architectures de réseaux sans fil à savoir : les architectures avec infrastructures et sans infrastructure (ad hoc) ensuite, nous détaillons la deuxième architecture (ad hoc). Nous présentons quelques applications de cette dernière ensuite son mode de fonctionnement et les différents types de protocoles de routage.

2.2 Architectures des réseaux sans fil

Dans les réseaux sans fil,on distingue deux architectures : architecture avec infrastructure (ou avec point d'accès) et les architectures sans infrastructure (Ad hoc).

2.2.1 L'architecture avec point d'accès

Dans ce mode, on trouve une station particulière, appelée point d'accès, qui fédère autour d'elle les stations sans fil à portée radio. Ce point d'accès est généralement relié à un réseau filaire. Il permet aux stations sans fil de communiquer avec des stations du réseau filaire, il permet aussi à une station sans fil de communiquer avec une station sans fil, qu'elle dépende ou non du même point d'accès.

2.2.2 L'architecture Ad hoc

Dans ce mode, il n'existe pas de station particulière, le réseau fonctionne de façon totalement distribuée.Chaque station utilise une interface radio et peut transmettre des informations vers une autre station du réseau qui se trouve dans sa portée radio.Si un nœud veut transmettre des informations vers une destination qui ne se trouve pas dans sa portée radio, il doit utiliser d'autres nœuds intermédiaires pour router ces informations vers la destination.

Dans ce mémoire, nous nous intéressons beaucoup plus à ce type d'architecture, plus précisément nous attaquerons le problème de routage, dans ce chapitre, ensuite, dans le prochain chapitre, nous présentons des modèles de QoS dans ce type de réseaux.

2.3 Les Applications des réseaux sans fil

Les applications faisant recours aux réseaux sans fil couvrent un très large spectre, incluant les applications militaires et de tactique, les bases de données parallèles, l'enseignement à distance, les systèmes de fichiers repartis, la simulation distribuée interactive et les applications de calcul distribué.D'une façon générale, les réseaux sans fil sont utilisés dans toute application où le déploiement d'une infrastructure réseau filaire est trop contraignant, soit parce que c'est difficile à mettre en place, ou bien la durée d'installation du réseau ne justifie pas de câblage à demeure.Parmi les nombreuses applications, on cite les exemples suivants :

Gare ferroviaire : l'installation de câble conduirait à des saignées dans les dalles de béton et serait par conséquent onéreuse. Le réseau sans fil apporte clairement un avantage économique.

Transmission entre deux bâtiments : Dans certains cas, il est interdit de tirer un câble entre deux bâtiments, notamment s'ils sont situés de part et l'autre d'une voie de circulation. La solution du réseau sans fil peut procurer de substantielles économies.

Unité de maintenance industrielle : Considérons une unité de maintenance d'une usine chimique, d'une raffinerie, d'une centrale nucléaire ou d'un navire civil ou militaire. Cette unité doit pouvoir se déplacer partout et échanger des données avec les autres unités d'usine. Le réseau sans fil offre pour cela une solution intéressante, voire incontournable.

2.4 Normes des réseaux sans fil

La norme la plus dominante est la IEEE 802.11 et ses extensions (IEEE 802.11a, IEEE802.11b,...) qui est la référence pour plusieurs produits sur le marché. En plus de la norme IEEE 802.11, il existe la norme européenne HiperLAN (type 1 et 2) et la norme Bluetooth.

Dans ce mémoire, nous détaillons la norme 802.11car dans notre travail on s'intéresse seulement à cette norme.

2.4.1 La norme 802.11

La norme IEEE 802.11 définit les deux couches basses d'un réseau local sans fil : la couche physique et la couche liaison. Cette norme offre plusieurs variantes au niveau physique, tandis que la partie liaison est unifiée. Pour le niveau physique, la norme IEEE 802.11 propose trois couches physiques : DS (Direct Séquence), FH (Frequency Hopping) et IR (Infra Red). Elle dispose de deux modes de fonctionnement distincts, qui correspondent à des architectures différentes : le mode infrastructure avec points d'accès, qui sert surtout à connecter des stations IEEE 802.11 à d'autres types de réseaux, le plus souvent Ethernet, et le mode réseau Ad hoc, qui permet de créer des réseaux indépendants de stations IEEE 802.11.

La norme IEEE 802.11 met à profit deux techniques d'accès au canal de communication : l'accès à compétition appelé DCF (Distribution Coordination Function) et l'accès contrôlé PCF (Point Coordination Function).

Comme nous nous intéressons aux réseaux Ad hoc, et comme ces réseaux utilisent le mode d'accès DCF, nous détaillons ici ce mode

Le mode DCF

DCF : est une méthode d'accès basé sur la méthode CSMA (carrier sence multiple acces) utilisé dans les réseaux Ethernet. On lit souvent que la méthode que la méthode d'accès 802.11 est une technique CSMA/CA ou CA signifie collision Avoidance, c'est-à-dire évitement de collision. Mais il n'est pas possible d'éviter la collision pour deux stations qui choisissent le même instant de transmission. Dans les réseaux Ethernet, la technique CSMA est complétée par des méthodes de détection de collision. Mais ces méthodes ne peuvent pas être utilisées pour détecter la collision dans les réseaux sans fil

Pour résoudre ce problème, la norme 802.11 utilise autour de CSMA les techniques suivantes

- Un système CSMA qui forme le cœur de système d'accès.
- Un système d'accusé de réception.
- Un système de retransmission.

Le système CSMA

Le système CSMA est un système classique de détection de porteuse. Supposons qu'un nœud C veut transmettre un paquet, la station C observe si le canal est vide alors C peut transmettre. Si le canal est non vide (par exemple, un autre nœud A qui est entraîne de transmettre) donc le nœud C diffère sa transmission et attend la fin de transmission de A. Après la transmission de A, C attend une intertrame (destributed interframe spacing) et tire au sort une durée d'attente. Cette durée est calculée sous forme d'un nombre entier d'une durée élémentaire, appelée slot de collision. Ce nombre entier est tiré au sort entre 0 et un entier CW voir la figure 2.1.

L'attente d'un nœud qui rencontre le canal occuper est donnée par :

Attente = *alea_unif(CW)*durée de slot de collision.*

Où

alea_unif(CW) : donne une valeur entre 1 et CW.

Après l'intertrame qui suit la transmission de A, le nœud C attend que le canal soit libre pendant une durée correspondant au temps d'attente calculé.

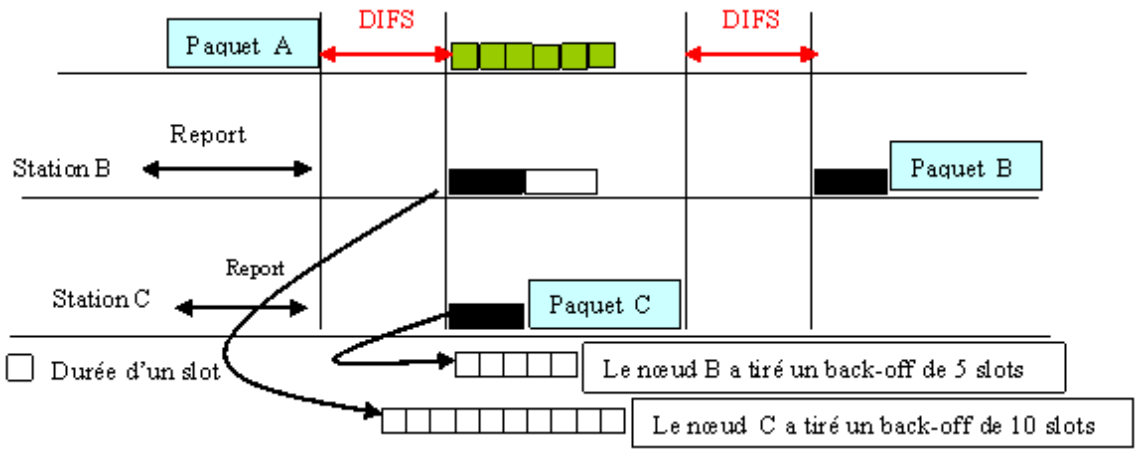

Figure 2.1 : Illustration de système CSMA

L'accusé de réception pour les trames point à point

La Norme 802.11 détecte la collision par absence de l'accusé de réception point à point. Dans 802.11, l'accusé est envoyé juste après la transmission. Pour cella 802.11 utilise une technique d'intertrame variable. Cette technique permet de varier la priorité d'accès au canal.

La norme utilise trois intertrames de taille différente :

- L'intertrame courte (SIFS : short inter-frame spacing).
- L'intertrame pour l'accès distribué (DIFS).
- L'intertrame pour l'accès contrôle (PIFS).

Ici nous présentons seulement les deux premiers (SIFS + DIFS) qui sont utilisés par DCF, voir figure 2.2.

De plus, un principe RTS/CTS (Request To Send / Clear To Send) est utilisé pour résoudre le problème des stations cachées. Avant d'émettre, une station envoie un message RTS pour réserver le canal, la destination répond par un message CTS s'il est prêt à recevoir. Ainsi, le canal est réservé pour la durée de la transmission. Les messages RTS/CTS et ACK sont prioritaires à l'accès au médium, car ils disposent d'un temps d'attente IFS (Inter Frame Space) inférieur à celui des paquets de données. Voir la figure 2.2

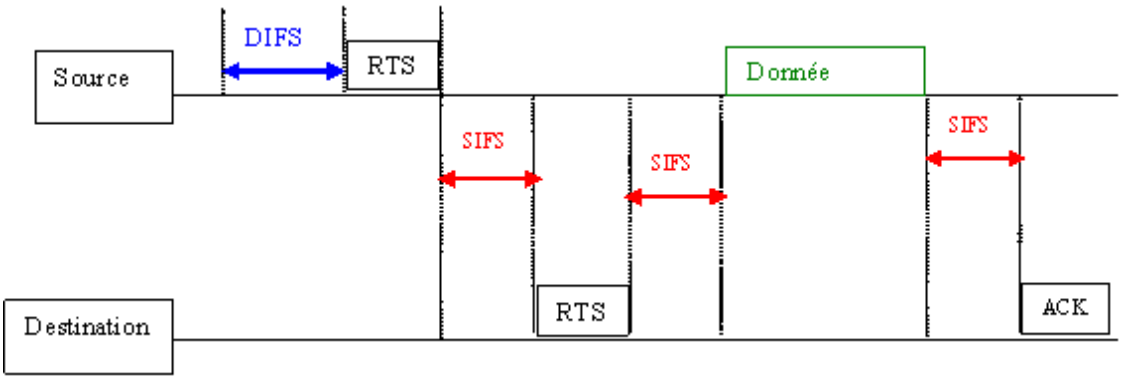

Figure 2.2 : Illustration du mode d'accès DCF

2.5 Les réseaux Ad Hoc

Les réseaux Ad hoc sans fil appelé généralement MANET (mobile ad hoc networks) sont considérés comme un système autonome dynamique composé des nœuds mobiles interconnectés par des liens sans fil. Les nœuds sont libres à se déplacer d'une manière aléatoire, s'organisent arbitrairement et la topologie du réseau peut être variée d'une manière très rapide et imprévisible. Voir (figure2.3 et figure2.4).

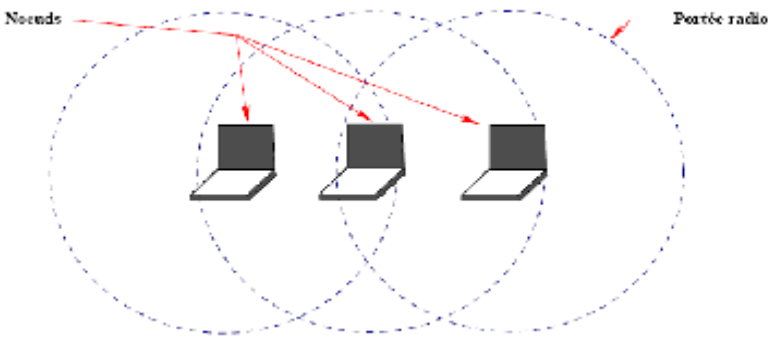

Figure 2.3 : Exemple du reseaux Ad hoc [38]

2.5.1 Caractéristiques des réseaux Ad hoc

Les réseaux Ad Hoc sont caractérisés par :

- **L'absence d'infrastructure** : chaque nœud travaille dans un environnement distribué. Chaque nœud peut agir comme un routeur pour relayer les communications [5].
- **La mobilité des nœuds et la maintenance des routes** : les nœuds peuvent se déplacer d'une manière libre et arbitraire. Ces mouvements créent des ruptures de chemins.

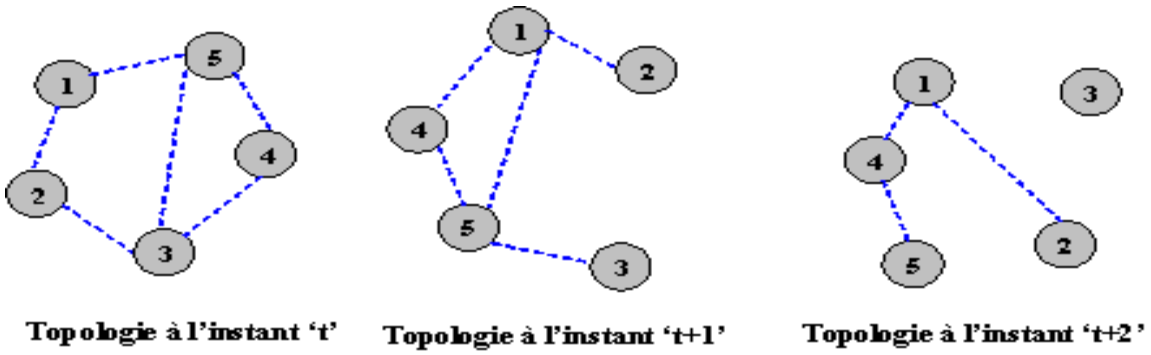

Figure 2.4 : Changement de la topologie dans les reseaux Ad Hoc

- **Hétérogénéité des nœuds** : un nœud mobile peut être équipé d'une ou de plusieurs interfaces radio ayant des capacités de transmission variées et opérant dans des plages de fréquences différentes. Les nœuds peuvent avoir des différences en terme de capacité du traitement (CPU, mémoire), de logiciel, de taille (petit, grand) et de mobilité (lent, rapide) [5].
- **Bande passante limitée** : dans les réseaux Ad hoc le medium est partagé entre plusieurs nœuds.
- **Contrainte d'énergie** : les nœuds mobiles sont alimentés par des batteries limitées, cela limite les applications supportées par chaque nœud.

2.6 Routage dans les réseaux Ad Hoc

Le routage désigne la façon de déterminer un chemin entre un émetteur et un récepteur pour transporter les paquets de données. On peut entrer ce chemin à la main dans les routeurs, cette méthode est appelée routage statique. Elle est envisageable dans les petits réseaux qui ne changent pas ou rarement. Mais elle devient impossible pour les grands réseaux ou pour les réseaux où la topologie est soumise à des modifications tel que les réseaux mobiles. Dans ce cas, les routes doivent être calculées en utilisant des protocoles de routage. Le but principal des protocoles de routage est l'établissement et la maintenance des routes, pour que les messages soient correctement délivrés dans le réseau. Plusieurs protocoles de routage ont été proposés, soit dans les réseaux filaires ou dans les réseaux sans fil. On distingue deux techniques de routage : le routage par vecteur de distance et le routage par état des liens.

Routage par vecteur de distance :

Dans ce type de routage, chaque nœud échange avec ses voisins une estimation de la distance vers tous les nœuds du réseau c'est-à-dire chaque nœud, diffuse à ses voisins directs (les voisins à un seul saut) sa table de routage à chaque modification de ce dernier, a des intervalles de temps fixes. Si un nœud ne reçoit pas la table d'un de ses voisins pendant une

période de temps fixée à l'origine, il considérera ce nœud comme défaillant. A chaque fois qu'une des stations reçoit une table de la part de son voisin elle rajoute les entrées qu'elle n'avait pas, d'autre part il compare le coût de ses chemins avec le coût des chemins reçus, si ces derniers sont moins coûteux, il modifie ses chemins. C'est-à-dire que chaque nœud ne conserve que la liste des nœuds du réseau et l'identité du voisin par lequel doit passer pour atteindre la destination par le chemin le plus court. A chaque destination possible est donc associée un next-hop[(1)] et une distance.

Ce type de protocole fait appel à l'algorithme BELLMAN-FORD pour le calcul des routes.

Routage par état des liens

Dans ce type des algorithmes, chaque nœud transmet, en diffusion dans le réseau, l'état des liens avec leurs voisins (diffuse les adresses de ses voisins et les distances qui le séparent a ses voisins). Par conséquent, tous les nœuds finissent par connaître tous les voisins de chacun des nœuds du réseau. Ensuite, il utilise l'algorithme de **Dijkstra** pour calculer les routes optimales entre un nœud source et une destination. L'algorithme de **Dijkstra** calcule les distances par une récurrence montante. En commençant par l'identification des voisins qui sont à distance 1, ensuite les voisins de ces voisins, c'est-à-dire les voisins à distance 2 et en continuant de telle sorte qu'on obtient une route optimale vers la destination.

Un des avantages de ce type de protocoles est sa capacité à pouvoir facilement trouver des routes alternatives lorsqu'un lien est rompu. Il est même possible d'utiliser simultanément plusieurs routes vers une même destination, augmentant ainsi la répartition de la charge et la tolérance aux pannes dans le réseau. En contrepartie, si le réseau est étendu, la quantité d'informations à stocker et à diffuser peut devenir considérable.

Dans les techniques de routage, on distingue généralement deux types de routage :

- **Routage à la source (source routing)**: consiste à indiquer dans le paquet routé l'intégralité du chemin qu'il doit suivre pour atteindre sa destination. L'en-tête du paquet contient la liste des adresses de différents nœuds relayeurs vers la destination.
- **Routage saut par saut (hop by hop)** : consiste à donner à un paquet uniquement l'adresse du prochain nœud vers la destination. Ce type de routage est plus répondu sur Internet.

2.6.1 Spécificité de routage dans les réseaux sans fil

Le routage dans les réseaux sans fil présente les particularités suivantes :

- Bande passante limitée des liens radios.
- Lien unidirectionnel.
- Caractère omnidirectionnel de la transmission radio.

[(1)] next hop ici designe le prochain saut

2.6.2 Evaluation d'un protocole de routage pour les réseaux sans fil

Pour évaluer les performances d'un protocole de routage, on recourt à des critères qualitatifs et quantitatifs pour déterminer si un protocole est bien conçu et s'il est performant ou non.

Les critères qualitatifs

On cite les critéres suivants:

- **Traitement distribué**

L'algorithme de routage doit être distribué et ne nécessite pas un nœud hôte qui assure la coordination entre les nœuds.

- **Absence de boucle en mode stationnaire**

Le calcul d'une route par le protocole de routage ne doit pas aboutir à des boucles. Dans le cas contraire, des paquets risqueront de tourner indéfiniment dans le réseau et consommer inutilement la bande passante.

- **Sécurité**

Sans certaines formes de sécurité au niveau de la couche réseau ou physique, un protocole de routage est vulnérable à plusieurs types d'attaques. Il est assez facile d'écouter le trafic, de rejeter les transmissions, de manipuler les en-têtes des paquets ou rediriger les messages de routage dans un réseau sans fil n'ayant pas des dispositions appropriées de sécurité.

- **Traitement des périodes de sommeil**

Le protocole doit permettre des périodes de sommeil pour certains nœuds pour conserver leurs énergies ou parce qu'ils sont parfois inactifs. Les nœuds du réseau peuvent s'arrêter parfois d'émettre ou de recevoir des informations

Critères quantitatifs

Les critères quantitatifs sont principalement les suivants :

- **Charge**

La charge générée par les paquets de contrôle de topologie du réseau est un critère très important puisqu'elle n'est pas utile à la véritable transmission d'information. Donc, un bon protocole ne génère pas une grande charge pour le contrôle.

- **Pourcentage des paquets perdus**

Le pourcentage des paquets perdus, à cause de la non disponibilité de la route, est un critère qui mesure la capacité d'un protocole à fournir dans toutes les conditions une route vers la destination.

- **Délai d'acheminement de bout en bout**

C'est un critère important à tout système de communication et en particulier les communications multimédias plus le delai est petit plus le protocole est efficace.

- **Le pourcentage de réception des paquets dans le mauvais ordre**

L'augmentation du pourcentage des paquets arrivant dans le mauvais ordre diminue l'efficacité du protocole de routage.

2.6.3 Les protocoles de routage dans les réseaux Ad hoc

Nous avons adopté l'approche du groupe MANET (Mobile Ad hoc Network) créée par l'IETF pour classifier les protocoles de routage dans les réseaux sans fil Ad- Hoc. Un réseau MANET est constitué de nœuds mobiles disposant de fonctionnalités de routage. Ces nœuds mobiles sont interconnectés par des liens sans fil, sans l'utilisation d'une infrastructure fixe et sans administration centralisée. Les nœuds sont libres de se déplacer aléatoirement et s'organisent arbitrairement. Par conséquent, la topologie du réseau peut varier de façon rapide et imprévisible. Un réseau ad hoc peut être autonome ou connecté à une infrastructure fixe.Plusieurs protocoles de routage ont été développés. Ces protocoles sont classifiés selon trois catégories comme suit:

2.3.3.1.Les protocoles réactifs

Ces protocoles ne cherchent à calculer une route qu'à la demande d'une application. L'établissement d'une route entre un émetteur et un récepteur s'effectue par inondation (voir figure 2.5). La Source diffuse un paquet à ses voisins, chaque mobile recevant ce paquet le retransmet si ce n'est pas déjà fait. Un numéro de séquence est utilisé pour éviter les boucles

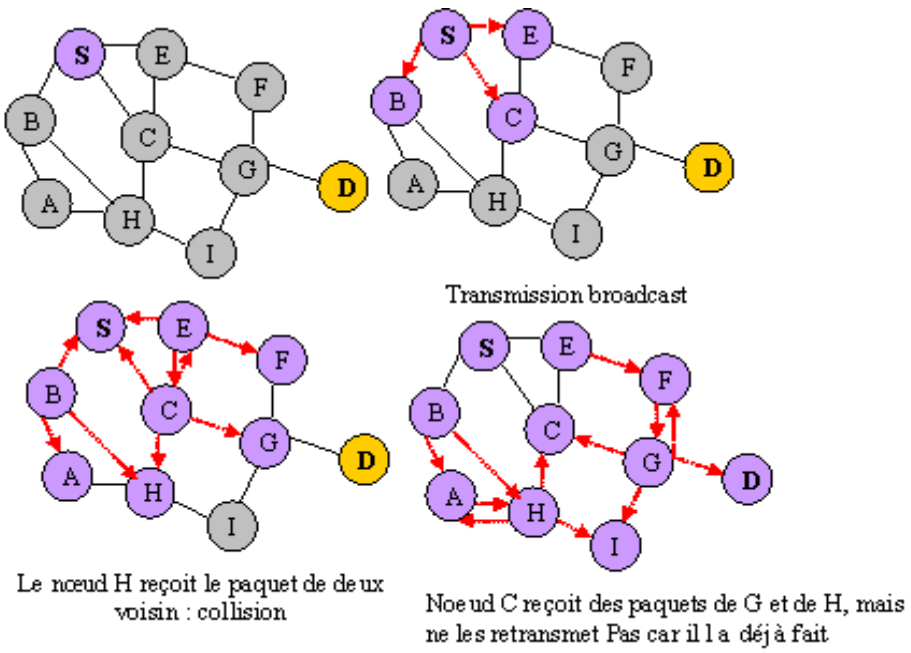

Figure 2.5 : Inondation

Comme l'établissement de la route est seulement à la demande alors le trafic est réduit. Par conséquent, la bande passante n'est essentiellement utilisée que pour la transmission de données.L 'inconvénient de ces protocoles est le délai d'acheminement qui est très important à cause de l'opération d'inondation.

Les principaux protocoles réactifs sont

- AODV (Ad-hoc on demand Distance Vector Routing).
- DSR (Dynamic Source Routing).

Le protocole AODV (Ad hoc on demand Distance Vector Routing)

Le protocole AODV [35],[41] est un protocole de routage réactif de type vecteur de distance. Il est exécuté en deux étapes:

Création de route

Lorsqu'un nœud reçoit une demande d'établissement d'une route vers une destination à partir d'une application, il cherche d'abord à savoir s'il en possède une. Si tel est le cas, le protocole n'a pas un algorithme particulier à mettre à jour et ce nœud va répondre à l'application. Dans le cas contraire, le nœud source diffuse un message de recherche de route (RREQ : Route Request) à travers le réseau, ce message est relayé nœud par nœud. Chaque nœud, recevant ce paquet, le diffuse à son tour jusqu'à ce qu'il atteigne la destination recherchée ou un nœud qui possède une information de routage plus fraîche vers la destination recherchée. Les nœuds intermédiaires conservent dans leurs caches l'adresse du nœud qui leur a relayé la requête. L'adresse de ce nœud fournit l'adresse du saut suivant dans le cas de la route vers la source pour le paquet RREP. La figure 2.6 (a), montre le mécanisme de création de la route. La source S diffuse une requête RREQ, ensuite chaque nœud recevant cette requête la diffuse jusqu'à ce qu'elle atteigne la destination D. Après, la destination D envoie une réponse RREP vers la source (voir figure 2.6 (b)).

Afin de limiter le coût dans le réseau et ne pas le surcharger avec des paquets diffusés, la recherche de la route dans AODV utilise une technique dite "Expanding Ring" qui limite le nombre de sauts pour les paquets de recherche de route à partir du nœud source. Cette technique commence par la recherche des routes dans le voisinage immédiat (a un seul saut) puis augmente peu à peu la distance de recherche si aucune route valide n'est trouvée. C'est-à-dire la requête est diffusée à un nombre de sauts limités. Si la source ne reçoit aucune réponse après un délai d'attente déterminé, elle régénère un autre message de recherche en augmentant le nombre de sauts. En cas de non-réponse, cette procédure sera répétée plusieurs fois avant de déclarer que la destination est injoignable.

Maintenance de la route

En cas de rupture d'un lien d'une route active, AODV essaye de réparer ce dernier, le nœud situé en amont de rupture diffuse une requête de recherche de route dans le voisinage. Si cette opération a échoué, il envoie un message d'erreur vers la source et la route sera

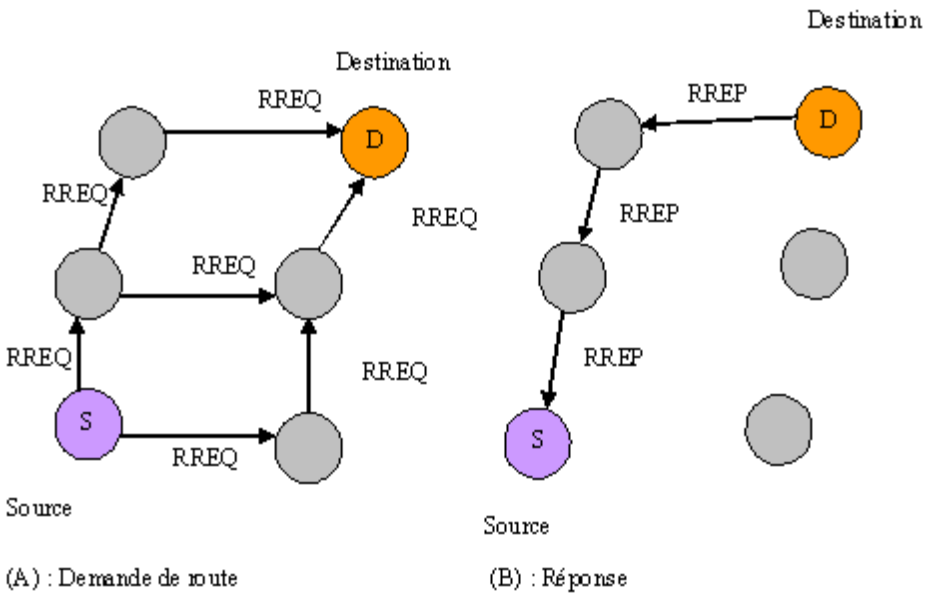

Figure 2.6 : La recherche d'une route

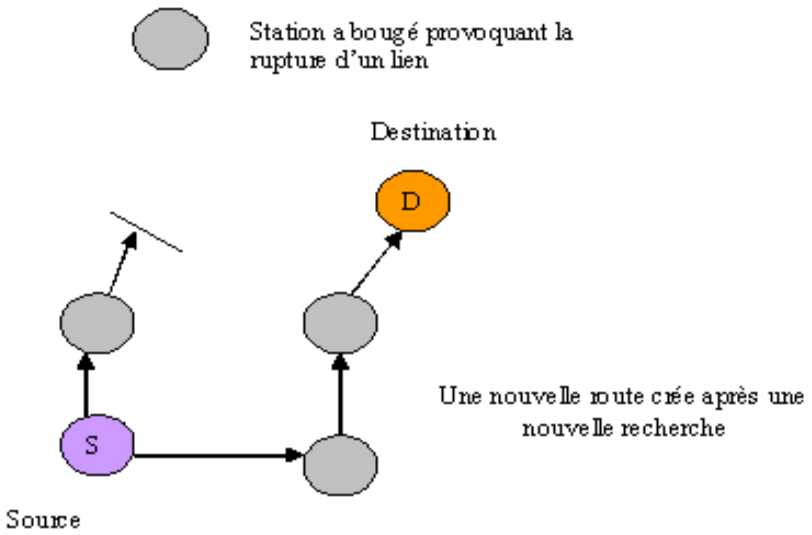

Figure 2.7 : Mécanisme de récupération d'une route avec AODV

supprimée des tables de routage de tous les nœuds intermédiaires. Ensuite, une nouvelle recherche de route sera lancée par la source [38]. (Voir la figure 2.7).

La table de routage de chaque nœud contient les informations suivantes :

- L'adresse IP de la destination.
- Le numéro de séquence pour la destination. Il permet d'utiliser les routes les plus frâıches.
- Le nombre de sauts pour atteindre la destination.
- Le prochain voisin désigné pour relayer le paquet vers la destination.
- La durée de vie d'une route : le temps pour lequel la route est considérée valide.
- La liste des voisins actifs. Un voisin est considéré actif s'il délivre au moins un paquet de donnée sans dépasser une certaine période.
- Le buffer de la requête.

Le protocole DSR (Dynamic Source Routing)

DSR [37], [42], [43] est un protocole réactif qui utilise une technique de routage par source, c'est-à-dire que le chemin à parcourir par le paquet de données est inclus dans l'en-tête du paquet. Le routage s'effectue de nœud à nœud en consultant l'en-tête du paquet. Les nœuds intermédiaires ne doivent pas nécessairement garder la trace de la route. Cela permet de résoudre le problème des boucles. Le DSR se compose essentiellement de deux mécanismes : le premier pour la création de la route, le second pour la maintenance de cette route.

Création de la route

Si un destinataire est dans le cache du nœud source alors la route est connue. Sinon une procédure de découverte de route est déclenchée. La source diffuse à travers le réseau un message de recherche de route (RREQ: Route Request) qui sera relayé saut par saut en rajoutant à chaque fois dans le message l'identifiant (l'adresse) du nœud courant (c'est-à-dire chaque nœud qui reçoit, le paquet ajoute à la route préexistante dans le paquet sa propre adresse) jusqu'à atteindre la destination ou un nœud qui possède une route vers la destination. Dans ce cas, ce nœud envoie une réponse (RREP: Route Reply) vers la source. Le paquet RREP contient la séquence des nœuds à travers lesquels la destination peut être atteinte. Cette réponse est envoyée en utilisant la route inverse créée par les nœuds traversés par la requête (RREQ).

La figure 2.8 montre la procédure de découverte de la route du nœud source 1 vers le nœud destination 8. Le nœud 1 envoie en broadcast une requête RREQ qui contient

- L'adresse de la source (Dans la figure 2.8 l'adresse du nœud 1).
- L'adresse de destination (Dans la figure 2.8 l'adresse du nœud 8).
- Un numéro d'identification unique qui permet aux nœuds intermédiaires de traiter la requête (RREQ) une seule fois.

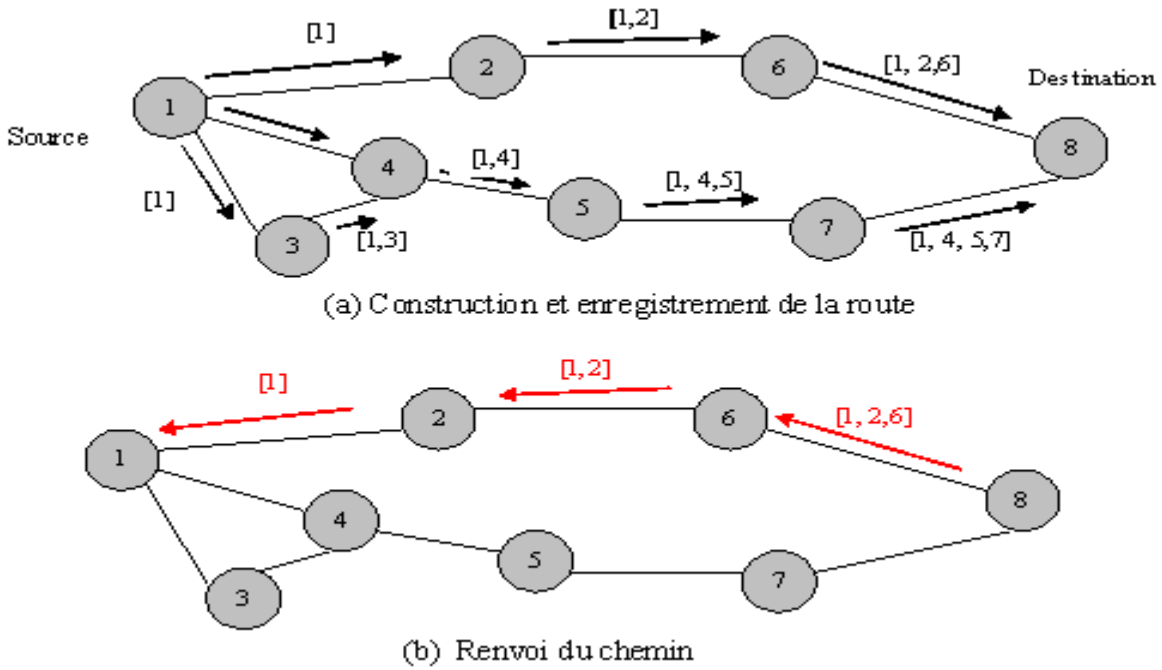

Figure 2.8 : Découverte de la route dans DSR

Chaque nœud intermédiaire ajoute son adresse dans le paquet RREQ. Ensuite, diffuse ce paquet à ses voisins. A l'arrivée de RREQ vers le nœud numéro 8, ce dernier génère une réponse RREP. Le paquet RREP contient la route déterminée par RREQ inversé (dans la figure2.8, RREP envoyé par la destination contient [6,2,1]). D'autre part, si un nœud intermédiaire dispose d'une route vers la destination, il copie sa cachette dans RREQ et génère une RREP vers la source.

- Si le réseau dispose des liens symétriques, le nœud (le nœud qui a répondu à la requête) inverse la route qui se trouve dans RREQ pour envoyer RREP vers la source.
- Si les liens ne sont pas symétriques, le nœud vérifie s'il dispose d'un itinéraire vers la source, il l'emploie, sinon il initialise une nouvelle demande de découverte de route vers la source.
- A la réception de RREP par la source, la source enregistre l'itinéraire porté par RREP dans sa cachette pour une future utilisation.

Entretien de la route

En cas de rupture d'un lien, le nœud situé en amont de la rupture envoie un message d'erreur RERR (le message contient l'adresse de nœud qui détecte l'erreur et le nœud qui le suit vers la source). Chaque nœud recevant ce message d'erreur supprime de sa cachette tous les chemins passant par le nœud qui a provoqué l'erreur. Ensuite la source initialise une nouvelle recherche de route.

Les nœuds intermédiaires peuvent garder dans leurs caches des routes crées par la procédure décrite précédemment. Ces routes permettent de répondre plus rapidement à d'autres requêtes de recherche de route. En cas de rupture d'un lien, si un nœud possède une route valide vers la destination, il peut détourner le trafic par cette nouvelle route.

2.3.3.2.Les protocoles proactifs

Le principe des protocoles proactifs est de maintenir à jour les tables de routage, de sorte que lorsqu'un mobile[(1)] désire envoyer un paquet de données à un autre mobile, une route soit immédiatement connue. C'est-à-dire que ces protocoles disposent des routes immédiatement vers les destinations joignables du réseau lorsqu'ils sont sollicités. Ceci est réalisé par le maintien continu des informations sur la topologie du réseau. Cette tâche est accomplie par un système d'échange périodique des paquets de contrôle de telle sorte que chaque nœud puisse construire d'une façon distribuée la topologie du réseau. On distingue entre deux types de paquets échangés :

- **Les paquets diffusés localement** : chaque nœud diffuse des paquets Hello qui lui permettent d'avoir une connaissance du voisinage. Ces paquets sont envoyés uniquement vers les voisins directs (à un seul saut).
- **Les paquets diffusés dans tout le réseau (les paquets de contrôle:TC) :** permettre à un nœud de diffuser l'état de son voisinage dans le réseau, ce qui permet à chaque nœud de connaître le voisinage des autres nœuds.

Dans la figure 2.9, le nœud numéro 5 diffuse des paquets hello vers ses voisins (3, 4, 6,7) et des messages de contrôle (TC) vers les autres nœuds du réseau tel que le nœud numéro 9.

Dans ces protocoles, chaque nœud maintient un ou plusieurs tables qui leurs permettent d'atteindre tous les autres nœuds du réseau. Lorsque la topologie évolue, les nœuds diffusent des messages de mise à jour à travers tout le réseau. Chaque nœud met à jour ses tables de routage lors de la réception des paquets de contrôle.

Les routes sont calculées par les algorithmes du plus court chemin tel que BELLMAN-FORD ou Dijkstra [39], [40].

L'avantage de ce type de protocole est la disponibilité immédiate des routes, quand les applications en ont besoin. Leur inconvénient est le coût du maintien des informations sur la topologie et de routage. Même en absence du trafic de données sur le réseau, le coût du maintien des informations de routage génère une consommation continuelle de la bande passante.

Les principaux protocoles de cette famille sont :

- OLSR (Optimized Link State Routing).
- FSR (Fisheye State Routing).

[(1)] mobile : en désigne ici par mobile un noeud de réseau

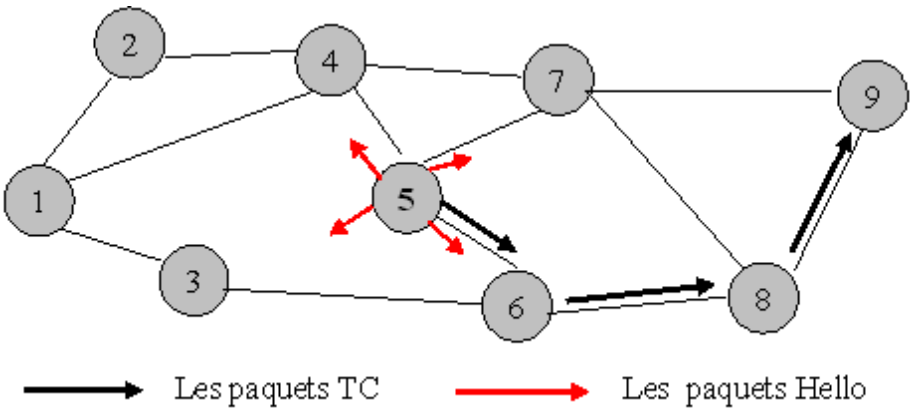

Figure 2.9 : Fonctionnement des protocoles proactifs

Le protocole OLSR (Optimized Link State Protocol)

OLSR [36] est un protocole de routage proactif qui fonctionne d'une manière distribuée. Il utilise un routage saut par saut et une technique de routage par état des liens pour calculer les tables de routage. Dans les protocoles de routage par état des liens, chaque nœud diffuse dans le réseau l'état des liens avec leurs voisins. Dans le cas du protocole OLSR, les nœuds ne diffusent qu'une sous-partie de leur voisinage grâce à la technique des relais multipoints.

Relais multipoints

Ils consistent essentiellement, en un nœud donné, à ignorer un ensemble de liens et de voisins directs, qui sont redondants pour le calcul des routes de plus court chemin : plus précisément, dans l'ensemble des voisins d'un nœud, seul un sous-ensemble de ces voisins est considéré comme pertinent. Il est choisi de façon à pouvoir atteindre tout le voisinage à deux sauts (tous les voisins des voisins), cet ensemble est appelé l'ensemble des relais multipoints (MPR: multi point relay).

Ces relais multipoints sont utilisés pour diminuer le trafic dû à la diffusion des messages de contrôle dans le réseau (échange d'état des liens, vérification périodique des liens...).

Diffusion par relais multipoints

La diffusion d'un message, à tout le réseau, par répétition, peut se faire par l'inondation classique en utilisant la règle suivante : un nœud retransmet un message si et seulement s'il ne l'a pas déjà reçu(il a entraîné de le recevoir pour la première fois).

La diffusion par relais multipoints diminue le nombre de retransmissions en utilisant les deux règles suivantes. Un nœud retransmet un message si et seulement si les deux conditions suivantes sont réalisés :

- Il ne l'avait pas déjà reçu (il a entraîné de le recevoir pour la première fois).
- Il vient de le recevoir d'un nœud dont il est un relais multipoint.

Ces MPRs seront les seuls nœuds à transmettre les messages de contrôles en diffusion (broadcast). Donc, plus le nombre des MPRs est diminué moins il y'aura de diffusion sur le réseau.

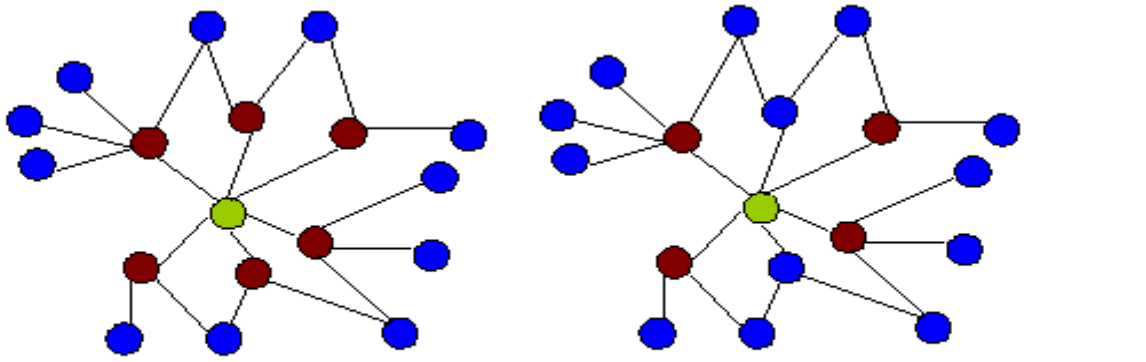

Figure 2.10 : Les relais multipoints

La figure2.10 donne un exemple de gain en nombre de retransmissions sur un graphe simple. Supposons qu'un nœud parmi les noeuds coloré en bleu dans la figure 2.10 émette un message vers le noeud central coloré en vert, dans la figure2.10 (a) pour transmettre ce message tous les voisins d'un noeud (les noeuds en Maron) retransmettent ce message, on a alors six retransmissions sont nécessaires. Par contre, en utilisant la retransmission par les relais multipoints seuls, les noeuds coloré maron dans la figure2.10 (b), on économise deux retransmissions. C'est-à-dire qu'on a quatre retransmissions seulement.

Pour maintenir à jour toutes les informations nécessaires pour le choix des relais multipoints (MPR) un nœud à besoin de connaître la topologie complète de son voisinage à deux sauts, cela est réalisé grâce à l'envoi périodique des paquets Hellos contenant la liste des voisins connus à un saut.

Le but des messages de contrôle est d'informer les autres nœuds des changements de la topologie pour qu'ils calculent les routes vers les autres nœuds. Ces messages contiennent l'adresse du nœud d'origine et la liste de ses MPRs. Quand le réseau est très dynamique, il est très intéressant de diminuer l'intervalle entre ces messages.

Le protocole FSR (Fisheye State Routing)

FSR [44],[45] est un protocole qui utilise la technique de routage par état des liens. Ce protocole part du principe qu'un changement de topologie lointain n'a aucune influence significative sur le calcul de la route localement. De ce fait, chaque nœud diffuse son voisinage local avec une fréquence qui dépend du nombre de sauts qu'un paquet doit effectuer. C'est-à-dire que plus un nœud est lointain moins il reçoit fréquemment les mises à jour de la topologie locale. FSR construit la table de routage en se basant sur des informations de la topologie et du voisinage.

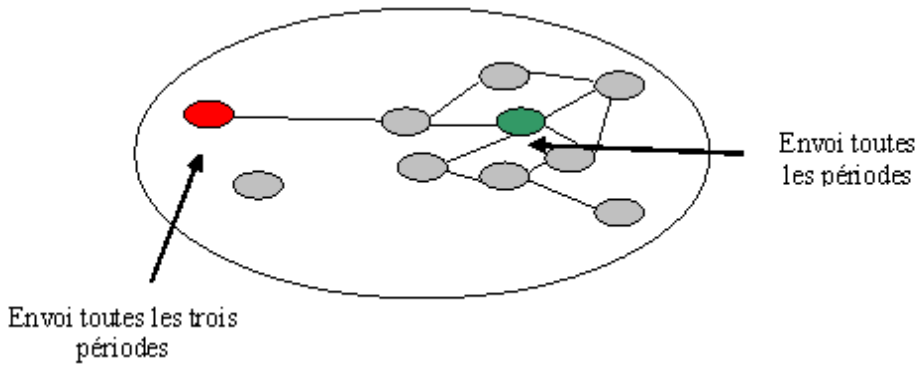

Figure 2.11 : Principe de fonctionnement de FSR

Tout nœud échange d'une manière périodique sa table de routage avec ses voisins.

L'information de routage devient de plus en plus précise au fur et à mesure que les paquets s'approchent de leurs destinations. A cause de cette propriété, FSR est souhaitable pour les réseaux à grande échelle.

2.3.3.3.Les protocoles Hybrides

Les protocoles hybrides combinent les deux idées des protocoles proactifs et réactifs. Ils utilisent une technique proactive pour avoir la route dans le proche voisinage (deux ou trois sauts).Au-delà de cette zone, les protocoles hybrides font appel aux techniques utilisées par les protocoles réactifs pour trouver la route. Dans ces protocoles, à la réception d'une requête réactive par un nœud, ce dernier peut savoir si la destination se trouve dans le voisinage (dans leur zone) ou non, et par conséquent il peut savoir aussi s'il faut aiguiller la requête vers d'autres zones sans déranger les autres nœuds de la même zone.

Zone routing protocol ZRP

ZRP est un protocole hybride. Il utilise une technique de routage proactif dans le voisinage proche et en dehors de cette zone de voisinage ZRP s'appuie sur une technique réactive. ZRP définit pour chaque nœud une zone de routage qui inclut les nœuds dans la distance (la distance est mesurée en nombre de sauts) minimale à ce nœud est x. les nœuds qui sont exactement à la distance x sont appelés les nœuds périphériques. Pour trouver une route vers des nœuds situés à une distance supérieure à x, ZRP utilise un système réactif qui envoie une requête vers tous les nœuds périphériques. ZRP met en œuvre deux types de fonctionnement : IARP [47] (IntrAzone Routing Protocol) et IERP [46] (IntErzone Routing Protocol).IARP [47] donne des routes jusqu'à une distance x et IERP [46] donne les routes vers des destinations dans la distance est supérieure à x.

Le processus de la recherche d'une route est le suivant :

- Si une route est connue, cela signifie que la destination est située à moins de x sauts.

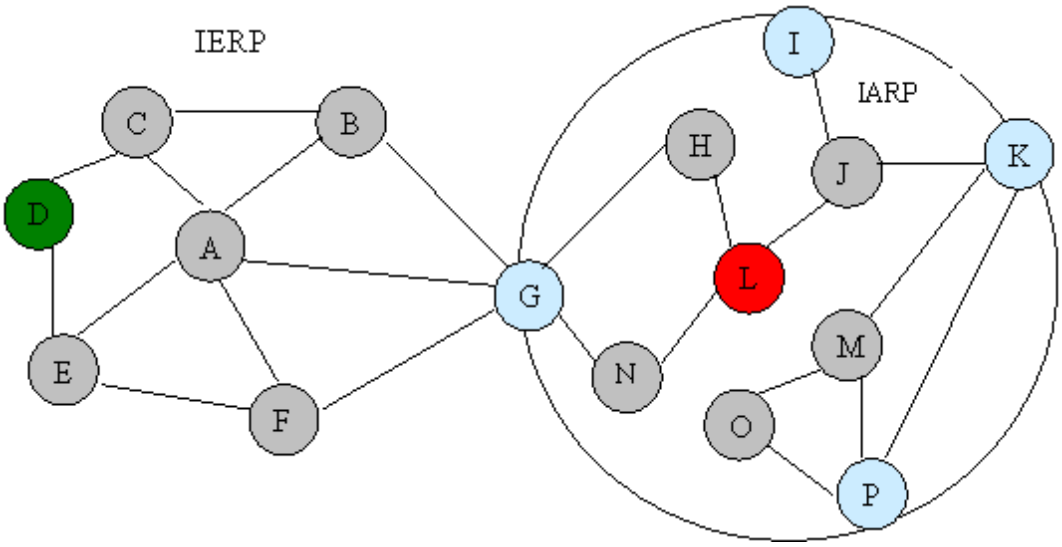

Figure 2.12 : Principe de fonctionnement de protocole ZRP

- Si aucune route n'est retournée donc le nœud destination est situé à une distance supérieure à x. Le nœud source envoie une requête par IERP vers tous les nœuds périphériques.
- Si un nœud périphérique a une connaissance d'une route disponible il renvoie une réponse.
- Dans le cas contraire, le protocole se poursuit récursivement jusqu'à l'obtention d'une route.

La figure 2.12 illustre le fonctionnement de ZRP avec une distance x = 2 sau

Supposons que le nœud L cherche une route vers le nœud D, D n'étant pas à une distance 1 ou 2 de L. Donc, une requête de demande de route est envoyée à tous les nœuds frontières (I, K, G, P) en utilisant IERP. Ces nœuds cherchent une route vers D de façon réactive. Une fois la route trouvée (dans cet exemple le nœud G qui trouve la route) elle est rapportée à L.

Le protocole CBRP (Cluster Based Routing Protocol)

Dans le "Protocole de Routage basé sur les Groupes " appelé CBRP (Cluster Based Routing Protocol) l'ensemble des nœuds du réseau est décomposé en groupes. Chaque groupe est constitué d'un représentant et des membres de groupes. Le principe de formation des groupes est le suivant : Un nœud p qui n'a pas de statut (c.-à-d. qui n'est ni membre ni représentant de groupe), active un timer et diffuse un message « Hello ». Lorsqu'un représentant de groupe reçoit le message « Hello », il envoie immédiatement une réponse à l'émetteur. Lors de la réception de réponse, le nœud p change son état « indécidé » à l'état « membre ». Si p dépasse un certain timeout en attendant la réponse et dans le cas où il possède un lien bidirectionnel vers au moins un nœud voisin, il déclare lui-même un représentant de groupe. Dans le cas contraire, p reste dans l'état « indécidé » et il répète la même procédure. A cause des changements rapides de la topologie des réseaux Ad hoc, l'attente des nœuds indécidés est très courte.

Afin de sauvegarder la répartition des nœuds dans les groupes, chaque nœud maintient une table des voisins. Chaque entrée de cette table est associée à un voisin, elle indique l'état du lien (unidirectionnel ou bidirectionnel) et le statut du voisin (membre ou représentant de groupe). Un représentant de groupe maintient les informations des membres qui appartiennent à son groupe. Il possède aussi une table des groupes adjacents. Une entrée dans cette table est associée à un groupe voisin, elle contient : l'identificateur du groupe et l'identificateur du nœud de liaison à travers lequel le groupe peut être atteint (voir la figure 2.13).

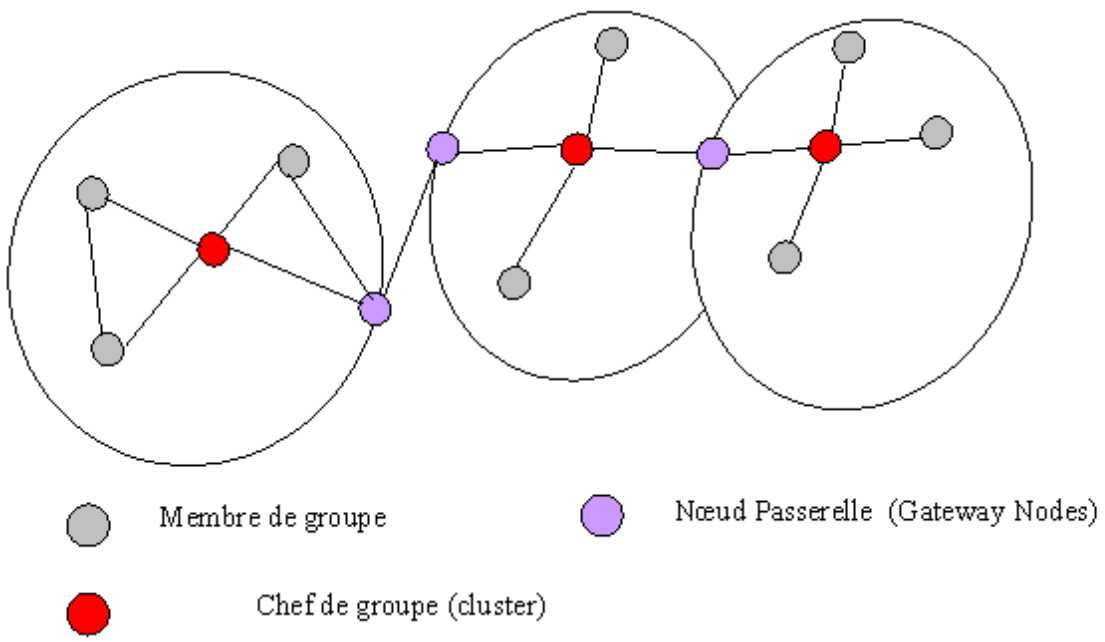

Figure 2.13 : Le protocole CBRP

Le routage dans le protocole CBRP se fait de la manière suivante : quand un nœud source veut envoyer des données à un nœud destination, il diffuse par inondation une requête de demande de chemin, et cela uniquement aux représentants des groupes voisins. Un représentant de groupe, qui reçoit la requête de demande, vérifie en utilisant sa table de membres de groupes - l'existence du nœud destination dans son groupe. Si la destination existe, le représentant y envoie directement la réponse, dans le cas contraire, la requête est diffusée aux représentants des groupes voisins. L'adresse des représentants des groupes est incluse dans la requête de demande de chemin, un représentant de groupe ignore toute requête déjà traitée. Quand la destination reçoit le paquet contenant la requête, elle répond par l'envoi du chemin qui a été sauvegardé dans le paquet de la requête. Dans le cas où le nœud source ne reçoit pas de réponse en expirant une certaine période, il envoie de nouveau une requête de demande de chemin.

Lors de l'acheminement des données, si un nœud détecte qu'un lien est défaillant, il fait retourner un message d'erreur à la source et applique un mécanisme de réparation locale. Dans ce mécanisme, si un nœud p trouve qu'un nœud suivant n, ne peut pas être atteint, il essaie de vérifier si le nœud n ou le nœud qui vient après n peuvent être atteints à travers un autre nœud voisin. Si l'un des deux cas est vérifié, les données sont envoyées en utilisant le chemin réparé.

2.6.4 Influence de type de protocole sur la transmission de la voix

Dans [6], les auteurs, S.Armenia, L.gallucio, A.leonardi et S.Palzarro ont fait une comparaison entre le protocole réactif AODV et le protocole proactif OLSR, ces auteurs ont démontré que les deux protocoles fournissent le même débit si on utilise les mêmes codecs. Les résultats obtenus par les auteurs sont donnés la figure 2.14.

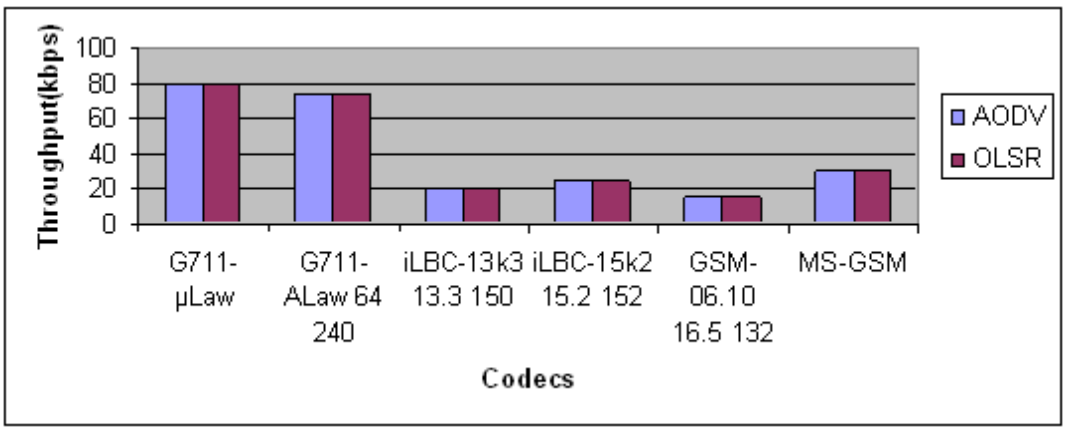

Figure 2.14 : Comparaison entre la transmission de la voix à travers AODV et OLSR en terme de débit

D'autre part, les meme auteurs ont démontré que le protocole proactif (OLSR) introduit un délai de bout en bout inférieur à celui introduit par le protocole réactif (AODV) voir figure 2.15. Ceci, car les protocoles proactifs peuvent délivrer les routes directement, car ces derniers maintiennent les routes dans leurs tables de routage contrairement au protocoles réactifs qui ne cherchent les routes qu'a la demande. Pour cella on peut dire que les protocoles proactifs sont plus adaptés pour la transmission de la voix.

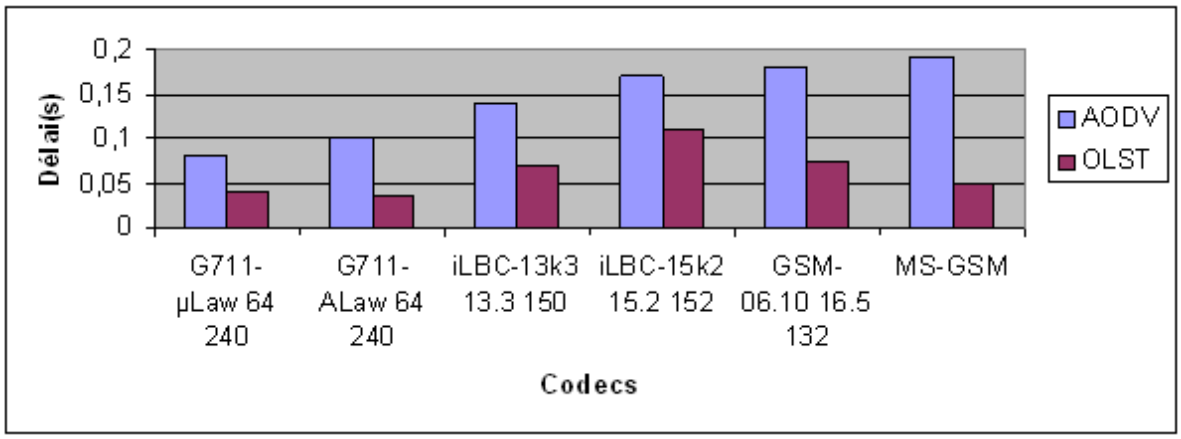

Figure 2.15 : Comparaison entre la transmission de la voix à travers AODV et OLSR en terme de délai

2.6.5 Routage multichemins

Le routage multichemins consiste à trouver deux ou plusieurs chemins entre un nœud source et un autre nœud destination.Il a été étudié dans les réseaux filaire pour :

- Augmenter la capacité globale du réseau.
- Un meilleur équilibrage de la charge.
- Redondance de chemin pour le rétablissement de la route en cas d'échec.

2.6.6 Les avantages de routage multichemins

Augmentation de la capacité

L'utilisation de multichemins permet une utilisation plus équilibrée des ressources du réseau. En effet, en répartissant le flux sur plusieurs chemins, on répartit par la même occasion l'utilisation des ressources des nœuds intermédiaires et le débit utilisé sur les liens. Sachant que les nœuds d'un réseau Ad hoc sont souvent limités en capacité de traitement, cette répartition est souhaitable. Par ailleurs, les capacités en débit disponible des liens ne sont pas indépendantes à cause de la nature du médium radio.

Augmentation de la fiabilité

A chaque changement d'état du réseau, des paquets de données sont susceptibles d'être perdus. L'utilisation de multichemins suggère deux solutions à ce problème. D'abord

implicitement, en répartissant les paquets successifs sur plusieurs chemins. En cas de rupture d'un des chemins, seuls les paquets émis le long du chemin défectueux peuvent être perdus. Ensuite explicitement, en émettant une copie du même paquet sur chacun des chemins possibles. Cette deuxième technique est toutefois inadaptée pour les protocoles qui ne gèrent pas les numéros de séquence au niveau de la couche transport (le protocole UDP par exemple).

Plusieurs protocoles multichemins ont été développés. Nous présentons ici le protocole TORA

Le protocole TORA

L'Algorithme de Routage Ordonné Temporairement ou TORA (Temporary Ordering Routing Algorithm) a été conçu principalement pour minimiser l'effet des changements de la topologie qui sont fréquents dans les réseaux Ad hoc. L'algorithme stocke plusieurs chemins vers une même destination, ce qui fait que les changements de topologie n'auront pas d'effets sur le routage des données, à moins que tous les chemins qui mènent vers la destination soient perdus (rompus). La caractéristique principale de TORA est que les messages de contrôle sont limités à un ensemble réduit de nœuds. Cet ensemble représente les nœuds proches du lieu de l'occurrence du changement de la topologie.

Dans ce protocole, la sauvegarde des chemins, entre une paire (source, destination) donnée, ne s'effectue pas d'une manière permanente. Les chemins sont créés et stockés lors du besoin. L'optimisation des routes (c.-à-d. l'utilisation des meilleurs chemins) a une importance secondaire, les longs chemins peuvent être utilisés afin d'éviter le contrôle induit par le processus de découverte de nouveaux chemins.

. TORA est un protocole basé sur le principe des algorithmes qui essaient de maintenir la propriété appelée « orientation destination » des graphes acycliques orientés (DAG : Directed Acyclic Graph). Un graphe acyclique orienté est « orientation destination » s'il y a toujours un chemin possible vers une destination spécifiée. Le graphe devient « non orientation destination », si un lien (ou plus) devient défaillant. Dans ce cas, les algorithmes utilisent le concept d'inversement de liens. Ce concept assure la transformation du graphe précédent, en un graphe orienté destination durant un temps fini. Afin de maintenir le DAG « orientation destination », l'algorithme TORA utilise la notion de taille du nœud. Chaque nœud possède une taille qui l'échange avec l'ensemble de ses voisins directs. Cette notion est utilisée dans l'orientation des liens du réseau. Un lien est toujours orienté du nœud qui a la plus grande taille vers le nœud qui a la plus petite taille.

Le protocole TORA peut être divisé en quatre fonctions de base : création de routes, maintenance de routes, élimination de routes et optimisation de routes.Chaque nœud i maintient un quintuplé qui lui est associé, ce dernier contient les champs suivants :

- t [i] : Le temps logique de défaillance.
- oid [i] : définissant le nouveau niveau de référence.

- r[i] : un bit indicateur de réflexion.
- delta[i] : le paramètre d'ordre de propagation.
- i: l'unique Identificateur du nœud.

Le protocole TORA permet la création de la route et la maintenance de cette dernière.

Création de la route

Le processus de création (ou de découverte) de routes pour une destination donnée, crée un DAG orienté vers cette destination. L'algorithme commence par initialiser : la taille (le paramètre d'ordre de propagation) de la destination à zéro et la taille des autres nœuds est indéfinie (c.-à-d. égale à NUL). Ensuite, le nœud source diffuse un paquet RREQ spécifiant l'identificateur de la destination, ID-destination, qui identifie le nœud pour lequel l'algorithme est exécuté.

A la réception de paquet RREP par un nœud.

- Si le nœud a une taille indéfinie (NUL), il rediffuse le paquet à ses voisins.
- Si le nœud a une valeur de taille différente de NUL, il répond par l'envoi d'un paquet UPD (update) qui contient sa propre taille.

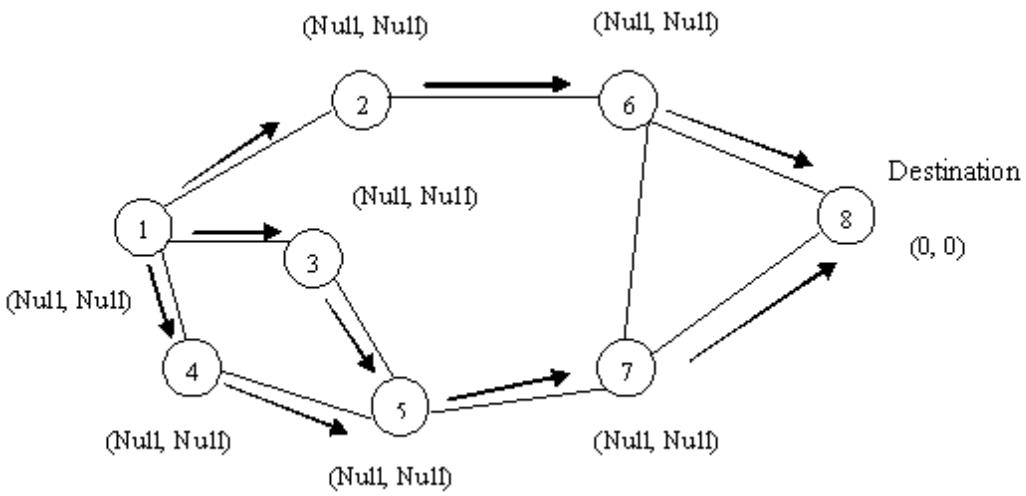

Figure 2.16 : Propagation de RREP dans le protocole TORA

Lors de la réception du paquet UPD, le nœud récepteur modifie sa propre taille comme suit : taille du nœud récepteur = valeur de taille contenant dans le paquet reçu + 1, à condition que cette valeur soit la plus petite par rapport à celles des autres voisins.

Par exemple, le nœud 6 de la figure 2.17 prend comme valeur de taille la plus petite taille des voisins (c.-à-d. la taille zéro qui correspond au nœud 8) plus un, ce qui donne une taille de 1. (La même chose pour les autres nœuds). De cette façon, un DAG est crée du nœud source vers le nœud destination.

Les figures 2.16 et 2.17, montrent la création de tel graphe dans le protocole TORA. Notons que les nœuds 5 et 7 reçoivent le paquet RREP deux fois, mais ils ne le diffusent

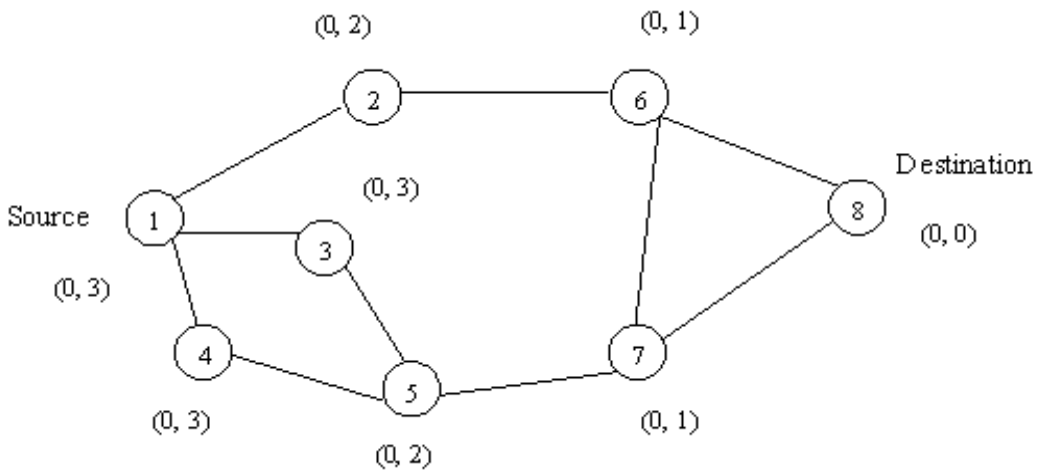

Figure 2.17 : Création des routes dans le protocole TORA

qu'une seule fois. Dans la figure 2.17, le lien, reliant les nœuds 6 et 7, n'est pas orienté, car les tailles des deux nœuds sont égales.

Maintenance de la route

A cause de la mobilité des nœuds dans les réseaux Ad hoc, des routes du DAG peuvent être rompues. Dans ce cas, une maintenance de routes doit être effectuée afin de rétablir un DAG pour la même destination. Quand un nœud i détecte une défaillance (sachant qu'il ne possède pas de suivants valides vers la destination), il lance un nouveau niveau de référence, son but est d'indiquer à la source l'invalidité du chemin. (voir figure 2.18).

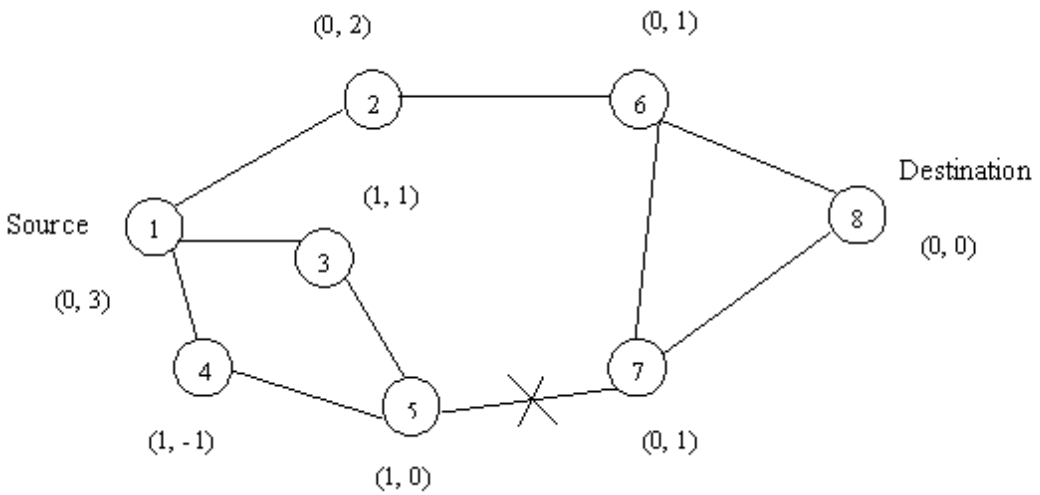

Figure 2.18 : Réaction du protocole TORA à la rupture d'un lien

La fonction de suppression du protocole TORA est effectuée en diffusant un paquet CLR (clear) dans le réseau, et cela afin de supprimer les routes invalides qui sont sauvegardées localement par les nœuds du réseau.

2.7 Conclusion

Nous avons présenté dans ce chapitre les types de routage dans les réseaux Ad hoc à savoir : le routage proactif, le routage réactif, le routage hybride et enfin le routage multichemins. Le routage proactif dispose des routes immédiatement vers les destinations joignables. Cette caractéristique, permet de réduire le délai d'établissement de la route du faite que la route est prise. Mais l'inconvénient de ce type de protocoles est la consommation de la bande passante à cause des messages de contrôle qui sont nécessaire afin de permettre à chaque nœud d'avoir une connaissance sur la topologie du réseau. Ces messages de contrôle peuvent augmenter aussi la charge du réseau et par conséquence, le nombre des paquets perdus augmente. Dans le cas où un protocole de routage réactif est utilisé, les routes sont créées à la demande, c'est-à-dire, quand un noeud veut initier une communication avec un autre noeud, il exécute un processus de découverte de route (Route Discovery). Une fois la route est trouvée, elle est maintenue par une procédure de maintenance de route, jusqu'à ce que la route ne soit plus utilisée. Cette méthode de calcule de route permet d'éviter le gaspillage de la bande passante et donc diminue la charge du réseau ce qui peut diminuer le nombre des paquets perdus par la congestion du réseau, mais d'une autre part elle augmente le délai d'établissement de la route.

Ces deux types de protocole ne prennent pas en considération le problème de qualité de service, car les routes ne sont pas choisies d'une manière à respecter le délai de bout en bout ou le taux de perte exigé par une application à temps réel par exemple la voix. Nous avons présenté aussi les protocoles de routage multichemins, qui offrent une solution partielle pour le problème de perte des paquets en répartissant les paquets successifs sur plusieurs chemins, en cas de rupture d'un lien, seuls les paquets émis le long de ce chemin défectueux seront perdus, mais ces protocoles peuvent augmenter la gigue du fait que les paquets sont transmis par des chemins différents ce qui est un effet négatif pour les applications à temps réel, et en particulier la voix.

Dans le prochain chapitre, nous présentons des protocoles de routages qui prennent en considération les problèmes de qualité de services et d'autres modèles de qualité de service dans les réseaux Ad hoc.

3

Les modèles de qualité de service dans les réseaux Ad hoc

3.1 Introduction

Dernièrement avec l'émergence des services multimédias à temps réel et l'augmentation des applications des réseaux Ad hoc, la qualité de service dans les réseaux Ad hoc est devenue un thème de recherche qui a pris beaucoup d'intérêts. Beaucoup de travaux ont été proposés pour l'introduction des applications multimédias dans les réseaux Ad Hoc. Cependant, il est très difficile de garantir une bonne qualité de service dans les réseaux Ad hoc car il faut prendre en considération les caractéristiques de ces réseaux à savoir : le changement dynamique de la topologie qui cause des ruptures dans les chemins, la bande passante limitée, l'énergie des nœuds.

Les modèles DiffServ et IntServ décrits précédemment (dans le chapitre 1) sont inadaptés pour les réseaux Ad Hoc du fait que la capacité des nœuds mobiles est limitée pour supporter des traitements complexes et gérer les réservations.

Plusieurs travaux ont été effectués pour améliorer la qualité de service dans les réseaux Ad hoc. Dans ce chapitre nous présentons quelques uns.

3.2 Le model FQMM

FQMM est le premier modèle qui a été proposé pour fournir la QoS dans les réseaux Ad hoc. Il combine entre les propriétés des deux modèles filaires : IntServ et DiffServ pour fournir une méthode hybride qui offre un traitement par flux pour les trafics les plus prioritaires et par classe pour les autres flux [5].FQQM est désigné pour les réseaux Ad hoc de petite taille qui ne depassent pas 50 nœuds avec une topologie plate et non hiérarchique. Dans ce modèle, on distingue trois types de nœuds :

- Les nœuds d'entrées (ingress node) : ils permettrent de marquer, classifier et envoyer les paquets.

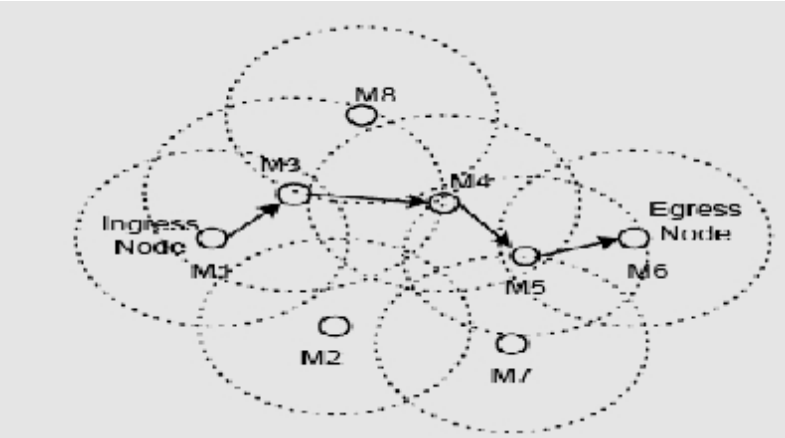

Figure 3.1 : Exemple de FQMM

- Les nœuds intermédiaires (interior nodes) : ils permettrent de transporter les paquets vers la destination selon les étiquettes attribuées par les nœuds d'entrée.
- Les nœuds de sortie (egress node) : sont les nœuds destinataires.

FQMM suppose que le protocole du routage fournit des routes ayant suffisamment de ressources.

L'avantage de ce modèle est la possibilité d'interfacer le réseau avec Internet, car il utilise des mécanismes de QoS plus proche que ceux utilisés dans les réseaux filaires.

Dans la figure 3.1, une route est établie entre le nœud d'entrée (ingress node) M1 et le nœud de sortie (egress node) M6. Les données sont émises de M1 vers M6. M1 est un nœud d'entrée (ingress node), il permet de classifier et de marquer les paquets. Les nœuds M3, M4 et M5 sont des nœuds intermédiaires (interior nodes) qui transportent les paquets de données, selon l'étiquette attribuée par le nœud M1, vers le nœud destinataire (egress node) M6.

Dans ce modèle, MANET représente un seul domaine DiffServ où le trafic est généré par une application qui s'exécute sur un nœud d'entrée M1 et se termine à un nœud de sortie M6.

3.3 Le model SWAN

Le modèle SWAN est basé sur les algorithmes de contrôle distribués afin d'assurer une différenciation de services dans les réseaux Ad hoc. Il offre la priorité pour le flux à temps réel en contrôlant la quantité de trafics best effort acceptés par un nœud. L'architecture SWAN est constituée de :

Un contrôleur d'admission : il permet d'accepter ou rejeter un nouveau trafic à temps réel.

Classificateur de flux : il permet de distinguer entre le trafic à temps réel, qui sera ensuite orienté directement vers la couche MAC, et le trafic best effort qui passe par le shaper (voir figure3.2).

Shaper : il permet de contrôler le trafic best effort. Son but est de retarder le trafic « best effort» dans la conformité avec le taux calculé par le contrôleur de taux « Rate controller » afin de donner une priorité pour le trafic à temps réel.

Rate Controller : il permet de contrôler le taux de trafic « best effort» en utilisant des informations sur le délai reçu à partir de la couche MAC. Le contrôleur de taux « Rate controller » détermine le taux de départ de Shaper en utilisant l'algorithme AIMD (Additive Increase Multiplicative Decrease) donné par l'algorithme 3.1 et les informations de délai retournées par la couche MAC.

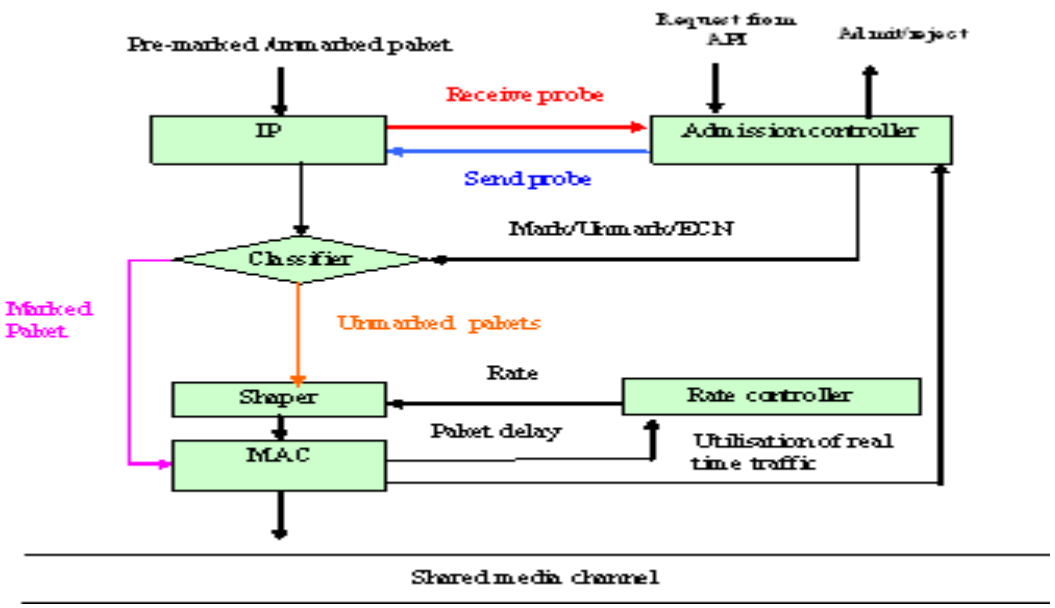

Figure 3.2 : Le modèle SWAN

SWAN utilise l'algorithme AIMD pour le contrôle du taux de départ de shaper. Chaque T seconds chaque nœud, incrémente son taux de transmission par C kbps jusqu'à ce que le délai mesuré par la couche MAC devienne excessif. Le module « rate controller » détecte un délai excessif, lorsque le délai d'un ou de plusieurs paquets dépasse un seuil d.

Le taux de départ de shaper est ajusté chaque T seconds. La période T doit être plus petite d'une manière qu'elle sera assez sensible à la mobilité du réseau Ad hoc. S'il y a une grande différence entre le taux de départ de shaper et le taux de transmission actuel, le nœud est capable de transmettre en rafale sans contrôle pour limiter le potentiel d'exécution du trafic à temps réel. Pour résoudre ce problème, le contrôleur de taux gère la transmission actuelle. Si la différence entre le taux de départ de shaper et le taux de transmission actuel est supérieur à g% du taux de transmission actuel alors le contrôleur du taux 'rate controller' ajuste le taux du départ de shaper d'une manière qu'il devient g% au dessus du taux actuel.

Algorithme 3.1 *algorithme AIMD /*appelé chaque T seconds */*

1**: **Begin
***2**: **If (n>0)** /* Si un ou plusieurs paquets ont un délai supérieur à un seuil d, n représente le nombre des paquets ayant un délai >d /*
***3**: S =S*(1- r /100) /* S: dénote le taux de sortie de shaper*/ /*Décrémentation multiplicative de taux de départ de shaper par r% */*
4**: **Else
***5**: S=S + C /* Incrementation de taux de départ de shaper par C kb */*
***6**: **If ((S- a) > a*g/100)** /* La différence entre le taux de départ de shaper et le taux de transmission actuel[1] de noeud est supérieur à g% */*
***7**: S =a*(1+g/100) /* Ajusté le taux du départ de shaper */*
8**: **End.

Pour accepter un nouveau flux à temps réel, le contrôleur d'admission diffuse une requête RREQ pour sonder la largeur de la bande passante disponible sur le chemin vers la destination. Le paquet RREQ contient un champ qui sera initialisé par la largeur de la bande disponible au niveau de la source. A la réception de ce paquet par un nœud intermédiaire (qui se trouve sur le chemin vers la destination), ce dernier compare la largeur de la bande disponible à son niveau avec ce champ. Si sa largeur de bande est inférieure à ce champ alors le nœud modifie ce champ en le remplaçant par la largeur disponible à son niveau (le champ prend la valeur de la bande disponible au niveau du nœud), ensuite diffuse le paquet dans le réseau. Lorsque la destination reçoit ce paquet, elle envoie une réponse vers la source qui contient la largeur de bande disponible sur la route (source- destination). Après la réception de la réponse, la source décide de l'acceptation ou le rejet du flux en se basant sur la largeur de bande disponible le long du chemin.

[1] le taux de depart actuel : est le taux de départ actuel de noeud vers les autres noeuds

En cas de congestion, les bits ECN (explicit congestion notification) de l'en-tête de la couche IP sont positionnés pour permettre à la source de ré-initier le contrôleur d'admission. Si la bande passante disponible ne satisfait pas les besoins du trafic alors le trafic est supprimé.

3.4 Les protocoles de QoS avec signalisation

Le but des protocoles de signalisation est de fournir un moyen de propager des informations de contrôle à travers un réseau. Les informations transmises peuvent être de différentes natures. Il peut s'agir d'informations topologiques, de demandes de recherche de routes satisfaisant certaines contraintes ou encore de rapports sur l'état du réseau et la disponibilité des ressources.

Les protocoles de qualité de service avec signalisation sont utilisés pour réserver et libérer les ressources. Deux mécanismes principaux doivent être inclus dans ces protocoles. Premièrement, l'information de qualité de service doit être sûrement diffusée entre les routeurs. Deuxièmement, l'information de signalisation de QoS doit être correctement interprétée et activée vers le nœud spécifié. La signalisation de QoS peut être divisée en deux catégories : signalisation 'in-band ' et signalisation 'out-of-band'.

Dans la signalisation 'in-band ', les messages de contrôle sont encapsulés dans les paquets de données IP, ce qui permet de réduire l'overhead généré par les messages de signalisation, contrairement à l'approche 'out-of-band' qui utilise des messages de contrôle explicite. Nous présentons ici le protocole INSIGNIA qui est un protocole de signalisation « in-band ».

3.4.1 Le protocole de réservation de ressources INSIGNIA

INSIGNIA est le premier protocole de signalisation spécialement conçu pour les réseaux Ad hoc par Ahn et Al en 1998. Il utilise la signalisation 'in-band', établit une réservation de la bande par flux. Il supporte deux types de flux : le flux à temps réel et le flux « best effort ».

INSIGNIA offre des mécanismes de réservation, restauration et adaptation pour répondre aux changements rapides de la topologie et les conditions de la qualité de service de bout en bout. La demande de réservation de la route est effectuée lors de l'envoi du premier paquet de données, les rafraîchissements s'effectuent par l'envoi périodique des messages de rafraîchissement.

Pour contrôler la qualité de service, le récepteur envoie périodiquement des rapports qui contiennent des statistiques sur le délai de bout en bout, la gigue, le taux de perte ainsi que l'état de la route vers l'émetteur.

Le paquet de la réservation de ressources se compose des champs suivants (voir la table3.1).

Reservation mode	Service type	Payload indicateur	Bandwidth indicateur	Bandwidth request
REQ/RES	RT/BE	BL/EL	Max/Min	Max/Min
1bit	1bit	1bit	1bit	16bits

Table 3.1: Constituants du paquet INSIGNIA

Mode de réservation (Reservation Mode)

Le mode de réservation est représenté par le bit : réservation mode. Quand un nœud veut établir une route vers une destination, il initialise le bit de reservation par REQ (voir table3.1), ensuite, il envoie le paquet vers la destination. A la réception de cette requête par les nœuds intermédiaires, ces derniers appliquent un contrôle d'admission pour décider de l'acceptation ou du rejet de la requête. L'acceptation ou le rejet d'un flux est effectué selon les ressources disponibles au niveau du nœud et les ressources spécifiées dans le champ : indicateur de la largeur de bande (bandwidth indicator). Si la requête est acceptée alors les ressources seront réservées et les paquets suivants seront programmés. Dans le cas contraire, les paquets seront traités selon le service « best effort». Si la destination reçoit un paquet avec le champ mode de réservation initialisé à RES, il envoie un rapport vers la source pour indiquer qu'une réservation de bout en bout a été établie avec succès

Type de service (service type)

Le type de service indique le niveau du service demandé dans la requête. Il indique un flux à temps réel si ce champ est initialisé par RT, et un flux « best effort» si ce dernier est initialisé par BE.

La requête de réservation de la bande passante (Bandwidth Request)

Permet à la source de spécifier sa largeur de bande maximum (max) et minimum (min) nécessaire pour le flux à temps réel.

Indicateur de la Charge utile (Playload Indicator)

Ce champ indique le type des paquets à transmettre. Le protocole INSIGNIA supporte deux types de paquets : des paquets avec une QoS base (BL), des paquets avec enhancement QoS (EL).

Indicateur de la largeur de bande (Bandwidth Indicator)

L'indicateur de la largeur de bande joue un rôle important pendant l'installation du flux. Il indique la disponibilité des ressources au niveau des noeuds intermédiaires. Si la destination reçoit un paquet de demande d'installation avec l'indicateur de largeur de bande

positionné à max c'est-à-dire que tous les noeuds en long de la route (les nœuds traversés par le paquet) ont une largeur de bande suffisante pour soutenir la demande spécifiée dans le paquet. En-cas contraire, au moins un nœud entre la source et la destination a des ressources insuffisantes pour répondre à l'exigence maximale de la largeur de la bande.

Installation du flux à temps réel

Le nœud source initialise le champ de réservation REQ, ensuite diffuse le message « reservation request » vers la destination. La requête est caractérisée par : le mode de réservation (REQ), le type de service (RT: Real Time) et une charge utile valide (BL : base level ou EL : enhancement level) et la largeur de la bande maximale et minimale nécessaire pour la transmission du flux. Quand une demande de réservation de ressource est reçue par un noeud destinataire, le module INSIGNA vérifie le statut d'installation. Le statut de l'installation du flux est déterminé en vérifiant les options d'IP dans le paquet de la demande de réservation. Si l'indication de la largeur de bande est positionnée à Max alors tous les noeuds entre la source et la destination ont alloué des ressources avec succès.Si l'indication de la largeur de bande est placée à Min ceci indique que seulement une base QoS est soutenue. Dans ce dernier cas le niveau du service des paquets reçus par la destination et dont le champ : charge utile est égale à EL sera renversé de RT (Real Time) à BE (Best Effort).

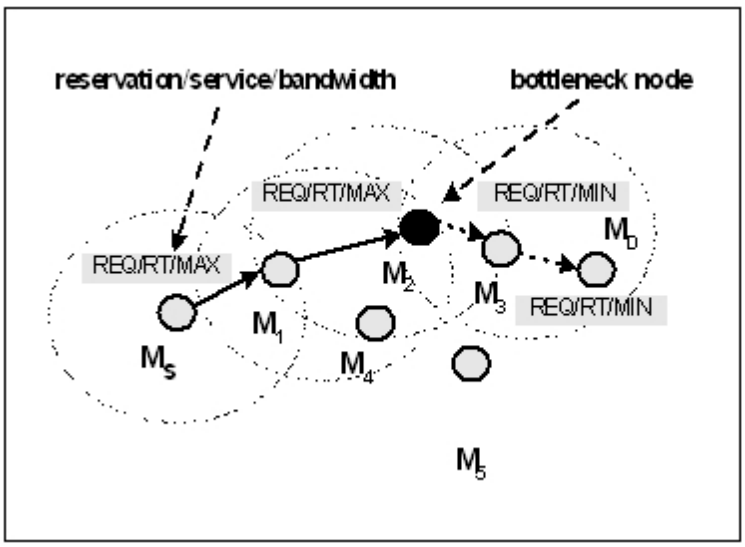

Figure 3.3 : Illustration d'installation de flux[24]

Dans la figure 3.3, le nœud source M_S initialise le champ max et envoie la requête vers le nœud destination M_D. Le nœud M_1 effectue un contrôle d'admission après la réception du paquet de la requête. Si les ressources spécifiées dans la requête sont disponibles à son niveau, alors M_1 alloue ces ressources. Ensuite le paquet (requête) est expédié vers le prochain noeud M_2.Ce processus est répété par chaque nœud jusqu'à ce que le paquet de

demande de reservation atteigne la destination M_D. Dans la figure 3.3, nous pouvons voir que entre M_2 et M_3 seulement la largeur de bande minimale est soutenue.Comme indiqué dans le champ « bandwidth indicateur ».A la réception de ce paquet, les noeuds, qui suivent M_2, refusent l'allocation des ressources.

Rapport de QoS

Ces rapports sont utilisés pour informer la source sur le statut des flux reçus. La destination envoie d'une manière périodique des rapports de qualité de service qui contiennent des statistiques sur (la perte, le débit, la largeur de bande, la gigue).

Restauration

Des flux sont souvent re-cheminés pendant la session à cause de la mobilité des noeuds. Le but de « flow restoration» est de rétablir la réservation aussi rapidement et efficacement que possible. La figure3.4 illustre le rétablissement de la route entre M1 et M3 après la migration du nœud M2, le noeud M1 agit sur le module de routage et diffuse un paquet sur une nouvelle route. A la réception de ce paquet par le noeud M4 le système INSIGNIA de M4 vérifie sa table d'état des flux. Si aucune réservation n'est trouvée pour ce flux alors le module INSIGNIA fait appelle au module « Admission control» et essaye de réserver les ressources pour ce flux.

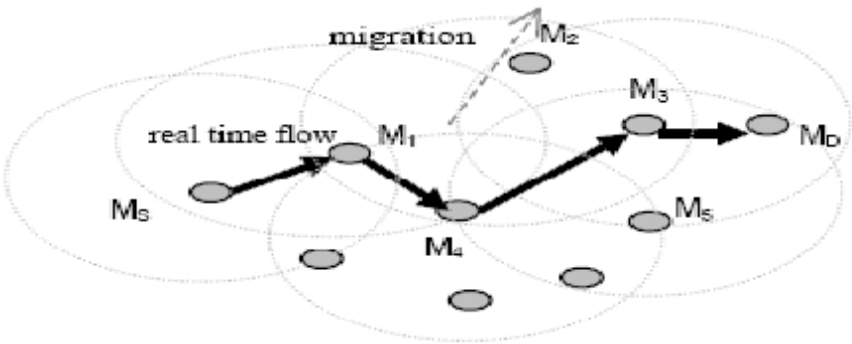

Figure 3.4 : Illustration d'une restauration

La figure 3.5 montre la position et le rôle du INSIGNIA dans le modèle de gestion du flux dans les routeurs des réseaux sans fil. Ce modèle est composé des modules suivants :

Signaling in-band (INSIGNIA) : permet d'établir, de terminer, de restaurer et d'adapter les réservations des ressources pour les flux à temps réel.

Packet forwarding : classifie les paquets reçus, ensuite diffuse ces paquets vers les modules appropriés (protocol routing, INSIGNIA, Adaptive mobile applications, packets sheduling, etc). Les messages de signalisation sont traités par le module INSIGNIA.

Routing protocol : mettre à jour la table du routage selon les changements de la topologie du réseau.

Admission control : il permet l'admission ou le rejet d'un flux à temps réel en se basant sur la largeur de la bande spécifiée dans la requête et la largeur de la bande disponible sur le canal local (au niveau du noeud).

Packet scheduling : il effectue l'ordonnancement des paquets.

Le module MAC : il contrôle l'accès au canal en tenant compte de la QoS.

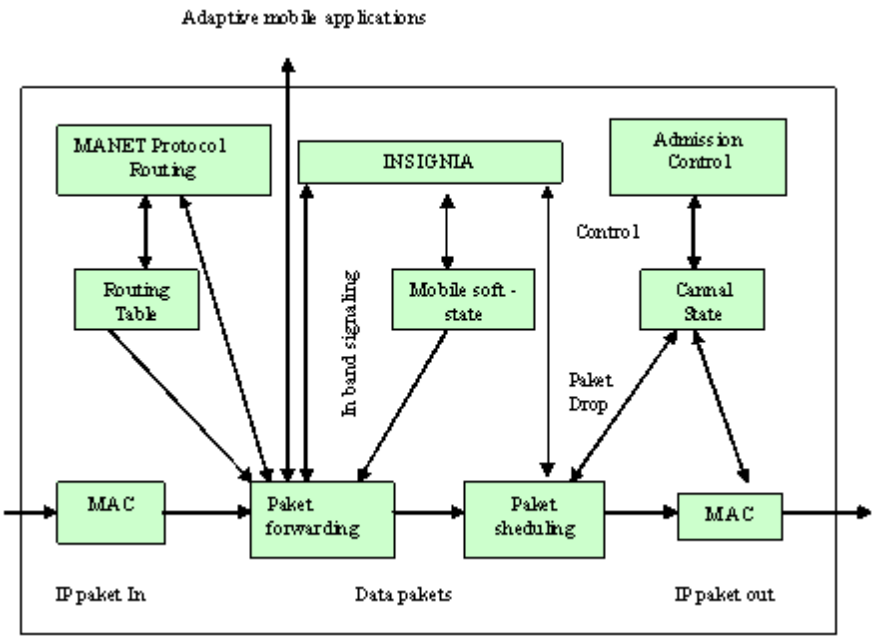

Figure 3.5 : Position de INSIGNIA dans le modèle de gestion du flux dans les routeurs

L'inconvénient de cette méthode est le fait d'avoir plusieurs informations sur le trafic dans chaque nœud ce qui pose un problème de capacité et de scalabilité avec l'augmentation du nombre des flux.

3.5 Les protocoles du routage avec QoS

Les algorithmes de routage traditionnels discutés dans le chapitre précédent ont été proposés pour router les données sans tenir compte des contraintes spécifiques ou à des demandes des utilisateurs. Ainsi, ils sont inadaptés aux applications qui nécessitent le support de la qualité de service.

Le principe des protocoles du routage avec qualité de service consiste à rechercher un chemin entre deux nœuds (source, destination) satisfaisant certaines contraintes. Plusieurs métriques peuvent être utilisées telles que le délai, la bande passante ou encore le coût de transmission. Le routage avec qualité de service ajoute en général à des protocoles de routage usuels un contrôle d'admission afin de sélectionner parmi les routes disponibles celles qui satisfont les contraintes du flux.

3.5.1 Le protocole CEDAR

CEDAR [21] est un protocole de routage réactif avec qualité de service basé sur une élection dynamique d'un coeur de réseau stable.Une grande partie de la largeur de la bande passante, est réservée pour l'échange des informations sur les liens stables entre les différents nœuds du réseau.Le calcul des routes est effectué par les nœuds du réseau cœur.

Le protocole CEDAR est utilisé dans les réseaux à taille moyenne (de 10 à 100 nœuds).Il se base sur trois étapes

Extraction d'un coeur du réseau

Un ensemble de noeuds est dynamiquement choisi pour calculer les routes et maintenir l'état des liens du réseau. L'avantage d'une telle approche est qu'avec un ensemble réduit de noeuds les échanges d'informations d'état et de route seront minimisés, évitant ainsi des messages supplémentaires circulant dans le réseau. En outre, lors d'un changement de route, seuls les noeuds du cœur serviront au calcul.

Propagation d'état de lien

Le routage avec qualité de service est réalisé grâce à la propagation des informations sur les liens stables avec une grande bande passante. L'objectif est d'informer les noeuds distants sur les liens de grande capacité, alors que les liens de faible capacité restent connus au niveau local (les noeuds n'ont pas une information sur la topologie globale du réseau).

Calcul de route

Celui-ci est basé sur la découverte et l'établissement d'un plus court chemin vers la destination satisfaisant la bande passante demandée. Des routes de 'secours ' sont utilisées lors de la reconstruction de la route principale, quand cette dernière est perdue. La reconstruction peut-être locale (à l'endroit de la cassure), ou à l'initiative de la source.

3.5.2 Le protocole des étiquettes basé sur le sondage

Compte tenu du coût d'accès au médium élevé dans les réseaux Ad hoc, la recherche de routes par inondation peut devenir très coûteuse. Le but de Ticket Based Probing est de limiter ce surcoût et de fournir des garanties de qualité de service. Ce protocole de routage a été conçu pour des réseaux dans lesquels la mobilité est suffisamment faible. pour ne pas poser de réel problème (scénario de type salle de conférence). La durée de vie des routes doit être grande devant le temps nécessaire à l'établissement ou à la restauration d'une route. Le protocole utilise une technique de réparation locale des routes.

La découverte de route est limitée, car l'émetteur va associer à une demande de route un certain nombre de tickets qui va limiter la diffusion des requêtes. Un noeud a la connaissance des caractéristiques des liens vers ses voisins immédiats grâce à la transmission périodique

de paquets de signalisation. Il peut donc ainsi sélectionner efficacement les voisins à qui transmettre les demandes de route. Plus un flux de données aura de contraintes, plus on associera de tickets à la demande correspondante. Deux problèmes sont étudiés : établir des routes, les plus proches de l'optimal possible, de moindre coût avec des contraintes de délai et établir des routes de moindre coût avec des contraintes de débit. Afin d'augmenter la probabilité de trouver une route, on utilise deux types de tickets : les tickets jaunes permettent de rechercher des chemins respectant la contrainte imposée et les tickets verts permettent d'obtenir des solutions de faible coût.

Malgré le fait que les noeuds ne connaissent que leur voisinage immédiat, Ticket Based Probing est efficace puisqu'il permet de trouver des routes avec une probabilité proche des algorithmes basés sur l'inondation du réseau et meilleure que des algorithmes recherchant un plus court chemin.

Beaucoup des recherches ont été effectuées aussi dans le secteur du cheminement. Ces travaux cherchent des routes qui répondent à certains critères (tel que le délai, la largeur de la bande) comme [13] [14] [15] [16] [17] [18] [19] et aujourd'hui ces recherches ont conduits à des protocoles qui sont considérés comme assez mûr pour faire face à des contraintes d'énergie et au changement rapide et fréquent de la topologie du réseau provoqué par la mobilité des nœuds. Parmi ces protocoles qui produisent un routage avec qualité de service, on trouve QoS-AODV [16] qui cherche le plus court chemin en terme du délai de bout en bout et n'est pas en terme du nombre de sauts. Le protocole MP-DSR [17] découpe le flux à plusieurs paquets ensuite envoie les paquets vers la destination en utilisant des chemins différents. Le lecteur peut trouver dans [20] et [21] d'autres travaux sur le routage avec qualité de service dans les réseaux Ad hoc.

D'autres travaux ont été effectués afin d'assurer une bonne qualité de service dans les réseaux Ad hoc qui sont connectés à des réseaux fixes. Par exemple dans [25] MC Domingo et D.Remondo ont proposé un modèle appelé DS-SWAN qui assure une qualité de service de bout en bout pour les flux à temps réel entre les nœuds du réseau Ad hoc et ceux du réseau fixe, dans [26] et dans le même but les mêmes chercheurs ont réalisé un autre travail appelé FA-SWAN.

3.6 Conclusion

Dans ce chapitre nous avons présenté quelques modèles qui permettent d'assurer la qualité de service dans les réseaux Ad hoc. Chaque protocole présenté ici ne traite qu'un aspect particulier de la transmission dans les réseaux Ad hoc. Le modèle FQMM est limité par la taille du réseau qui ne doit pas dépasser 50 nœuds. D'autre part, ce modèle introduit seulement une différentiation entre les différents paquets d'un flux, mais ne comporte pas des mécanismes qui assurent la transmission des paquets à temps réels comme la voix par exemple pendant le délai et avec un taux de perte exigés par l'application. Le modèle SWAN

utilise une seule route pour la transmission des paquets de la source vers la destination, donc en cas de rupture de cette route les paquets envoyés sont tous perdus. Donc le modèle SWAN peut assurer le délai de bout en bout exigé par la voix, car le flux voix n'est accepté par le nœud source que si les ressources exigées par cette application sont disponibles. Par contre, une rupture de route entre la source et la destination provoque une perte de tous les paquets envoyés. Le protocole INSIGNIA enregistre plusieurs informations sur chaque flux et au niveau de chaque nœud, ceci pose un problème de capacité et de scalabilité.

Le protocole de routage CEDAR permet de calculer le plus court chemin entre la source et la destination, ceci peut satisfaire le délai de bout en bout exigé par la voix d'autre part, la propagation des informations sur les liens stables permet de diminuer le nombre des paquets perdus. Mais son inconvénient est la taille qui ne peut pas dépasser 100 noeuds.

Dans le chapitre prochain, nous présenterons le protocole MRTP qui prend en considération la mobilité des noeuds par l'utilisation des protocoles de routage multichemins et un mécanisme de feedback.

4

Le protocole MRTP

4.1 Introduction

Dans les chapitres précédents, nous avons présenté des protocoles de routages multichemins. Ces protocoles permettent d'envoyer les paquets à travers des chemins différents (totalement ou partiellement disjoints), ce qui permet d'équilibrer la charge et de diminuer le taux de perte. Nous avons présenté le protocole RTP/RTCP qui utilise une technique feedback (envoi des rapports de QoS par les destinations vers la source afin que ce derniere puise s'adapter à ces conditions.).

Dans ce chapitre nous présentons le protocole MRTP qui mélange entre ces deux techniques.

4.2 Le protocole MRTP

MRTP (Multi-flow Realtime Transport Protocol): est un protocole de transport des flux multimédias à temps réel. Il utilise des chemins multiples. Le protocole MRTP est situé dans la couche transport et implémenté dans l'espace réservé pour l'utilisateur. MRTP et son compagnon MRTCP (Multi-flow Realtime Transport Control Protocol) fournissent l'appui essentiel pour le transport des flux à temps réel par des chemins multiples. Dans ce protocole, le flux est divisé en plusieurs paquets, ensuite ces paquets seront envoyés sur plusieurs chemins. Le protocole MRTP utilise des mécanismes pour la gestion de la qualité de service tels que : les horodateurs (timestamp), les numéros de séquence, des rapports qui portent des statistiques sur les paramètres de QoS (le délai, les pertes, la gigue).

Le protocole MRTP est basé sur deux protocoles :

- Le protocole RTP : (nous l'avons présenté dans le premier chapitre) est un protocole Multicast pour le transport du flux à temps réel sur Internet. Mais ce protocole seul ne supporte pas l'utilisation des flux multiples. Comparé avec RTP le protocole MRTP, fournit une grande flexibilité, il permet le partitionnement de données ensuite la dispersion de

ces données sur les différents chemins reliant la source et la destination. MRTP est plus souhaitable pour les réseaux Ad hoc où la mobilité des nœuds cause des ruptures de chemins

- Le protocole STCP (nous l'avons exposé dans le premier chapitre) est un protocole qui supporte le multi homing et multi-Streaming. STCP permet l'utilisation des interfaces réseau multiples ou Stream multiple avec une seule session STCP. Mais ce protocole n'est pas applicable directement pour le transport des flux multimédias, car il ne dispose pas des fonctions pour la gestion de la qualité de service tel que : l'horodateur et le service de rétroaction (QoS feedback).

La figure 4.1 illustre une session MRTP où des flux multiples sont utilisés. L'émetteur divise le flux à temps réel X [n] en plusieurs paquets X1 [n]. . . Xk [n]. Ensuite, chaque paquet est assigné à un ou plusieurs flux par l'allocateur du flux. Les paquets traversent des chemins partiellement ou entièrement disjoints vers le récepteur. Le récepteur rassemble les paquets reçus à travers les différents chemins en utilisant un buffer appelé « buffer resequencing » pour chaque flux. Les paquets sont mis dans le bon ordre en utilisant les horodateurs et les numéros de séquence et l'identificateur du flux porté dans les en-têtes des paquets.

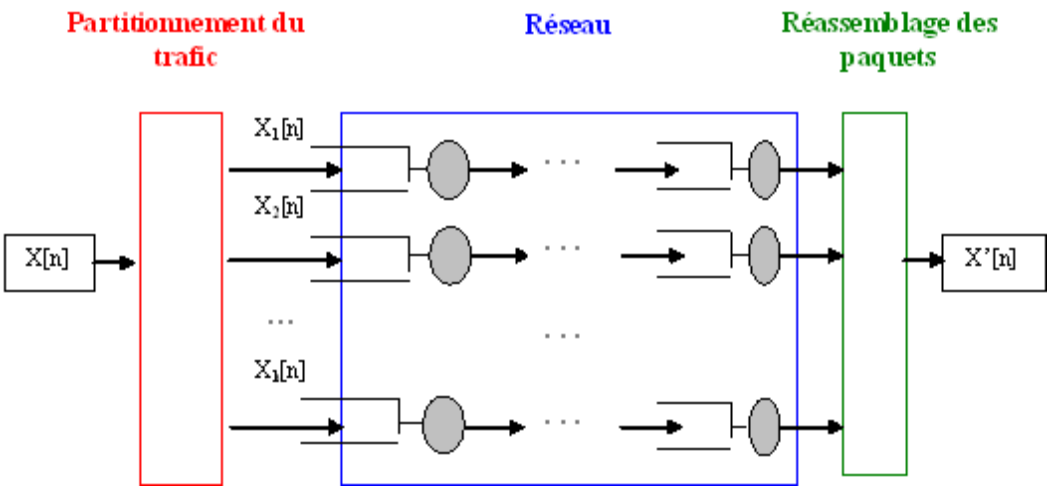

Figure 4.1 : Session MRTP[55]

La figure 4.2 représente la position du protocole MRTP/MRTCP dans la pile TCP/IP. Le protocole MRTP utilise le service datagramme d'UDP ou le service de transport de multihoming/multistreaming de SCTP.

Figure 4.2 : Position du protocole MRTP/MRTCP dans la pile TCP/IP[55]

4.3 Gestion des flux et Sessions

A la différence de RTP, MRTP est un protocole orienté session. Une session MRTP, devrait être établie d'abord entre l'émetteur et le récepteur par MRTCP avant de commencer le transfert des données. Dans cette étape, les deux nœuds d'extrémités (source et destination) échangent des informations telles que : les chemins disponibles, les identificateurs de session et du flux ainsi que le numéro de séquence initial.

Pendant la transmission des données, un nouveau flux peut être ajouté à une session quand un chemin est trouvé et un flux peut être supprimé de la session en se basant sur les rapports de QoS reçus à partir de la destination.

Une session a un identificateur unique ID généré d'une manière aléatoire, aussi chaque flux dans la session a un identificateur Id_flow unique généré aussi d'une manière aléatoire. Tous les paquets appartenant à une session portent l'identification de cette session et les paquets appartenant à un flux portent l'identification correspondante à ce flux. (C'est-à-dire que chaque paquet porte un identificateur de session et un identificateur de flux).

MRTCP définit un ensemble de messages pour la gestion des sessions et des flux.

4.4 Partitionnement du trafic

L'allocateur du trafic partitionne et disperse le trafic à temps réel à des flux multiples. Le schéma de base de la division et de la dispersion du trafic est affecté par un nombre de facteurs tel que : le nombre des chemins existant, les paramètres de QoS de chaque chemin à savoir le délai, la largeur de la bande,.... L'allocateur assigne les paquets aux différents chemins en utilisant l'algorithme de Round Robin. Cette opération de dispersion du flux permet un bon équilibrage de la charge.

4.5 Etablissement des routes pratiquement

Pour établir les routes entre une source et une destination le protocole MRTP se base sur les protocoles de routage multichemins de la couche Internet comme le protocole TORA ou le protocole DSR (voir chapitre 2).

4.6 Horodateur (Timestamping)

Cette fonction est semblable à celle de RTP [28]. L'horodateur porté par un paquet de données MRTP, dénote l'instant de la génération du premier octet de la charge utile. Le numéro de séquence indique le positionnement relatif de ce paquet dans l'ensemble des paquets assignés au même flux. A chaque flux est assigné un numéro de séquence initial. Ce numéro est généré d'une manière aléatoire pendant l'étape d'établissement de la session

ou quand un flux est ajouté à une session. Le numéro de séquence du flux est incrémenté par un à chaque transmission du paquet.

4.7 Rapports de QoS

Comme RTP, MRTP produit d'une manière périodique des rapports de QoS. Ces rapports portent deux types de statistiques : des statistiques pour chaque flux et pour chaque session. Dans RTP, les rapports de QoS sont transmis à un taux d'un rapport par T = max {Td, 5} second où Td est mis à jour d'une manière dynamique en se basant sur le nombre courant des participants et la largeur de la bande utilisée par la session. Cet algorithme maintient la largeur de la bande employée par les rapports de qualité de service (feedback) à un taux relativement constant par rapport à la largeur totale de bande utilisée par la session RTP. Bien que le nombre des participants peut changer pendant la session. Ces rapports de qualité de service (feedback) ne sont pas assez prompts pour l'expéditeur afin de s'adapter à la topologie très changeante dans le réseau Ad hoc, car le protocole RTP ne prend pas en considération la mobilité des noeuds.

A la différence de RTP, les rapports SR (Sender report) et RR (Receiver report) de MRTP peuvent être envoyés à des intervalles fixés par l'application.

4.8 Réassemblage des paquets au niveau du récepteur

Le récepteur emploie un buffer de réassemblage pour chaque flux MRTP, afin de contrôler et ordonner les paquets reçus, en utilisant les numéros de séquences qui se trouvent dans les en-têtes des paquets. Le flux de données à temps réel original est reconstruit en combinant les différents flux reçus et en utilisant les identificateurs des flux et les horodateurs inclus dans les en-têtes des paquets. Pour assurer une fiabilité du transport des données, les paquets arrivant tôt seront stockés dans le buffer temporairement, en attendant l'arrivée de tous les paquets dont le numéro de séquence est plus petit que celui du paquet reçu. D'autres parts, dans la plupart des applications à temps réel, un paquet est extrait à partir du buffer ensuite sera décodé même en absence d'un ou plusieurs paquets du même flux. Dans ce cas-ci, le décodeur peut appliquer les mécanismes de contrôle d'erreur (par exemple, FEC et MDC pour plus d'informations sur ces techniques voir [33]) et la dissimulation des erreurs (par exemple, remplacer le paquet manquant par son prédécesseur). Pour réduire les dommages provoqués par les paquets perdus.

4.9 Format des paquets MRTP/MRTCP

MRTP/MRTCP emploie trois types de paquets : des paquets pour la transmission des données, des paquets qui transmettent des statistiques sur la QoS (les rapports de qualité de service), des paquets pour le contrôle des sessions et des flux.

4.9.1 Les paquets de données

Le format d'un paquet de données est illustré dans la figure 4.3

<table>
<tr><td>Ver(4)</td><td>Pad(2)</td><td>Ext(1)</td><td>Mk(1)</td><td>Payload(8)</td><td>source ID(8)</td><td>Destination ID(8)</td></tr>
<tr><td>Session</td><td>ID</td><td>(16)</td><td>Flow</td><td>Id</td><td colspan="2">(16)</td></tr>
<tr><td colspan="2">flow</td><td>sequence</td><td>number</td><td>(32)</td><td colspan="2"></td></tr>
<tr><td colspan="3"></td><td>Timestamp</td><td>(32)</td><td colspan="2"></td></tr>
<tr><td colspan="4"></td><td colspan="3">Payload</td></tr>
<tr><td colspan="7"></td></tr>
<tr><td colspan="4">Payload</td><td colspan="3">Padding</td></tr>
</table>

Figure 4.3 : Format du paquet MRTP de données [55]

Version (Ver) : 4 bits, donne la version du protocole MRTP/MRTCP.

Padding (Pad) : 2 bits, représenté par le champ Pad quand ce dernier est égal à 1, cela indique que le paquet contient un ou plusieurs octets de bourrage (placer à la fin pour aligner les frontières de paquets avec un mot de 32 bits).

Extension (Ext.) : 1 bit, s'il est égal à 1 alors l'en-tête fixe est suivi d'un ou de plusieurs extensions.

Marker (Mk) : 1 bit, il est employé pour marquer des événements significatifs. L'interprétation de bit marqueur dépend du type de la charge utile. Il marque la fin d'armature pour la vidéo et début de talkspurt si les données sont de type audio.

Payload type : 8 bits, il porte une valeur qui indique le format de la charge utile. Il identifie le genre de la charge utile contenue dans le paquet (exemple : JPEG vidéo or GSM audio.).

Source ID: 8 bits, permet d'identifier l'émetteur du paquet pendant une session. Il est généré aléatoirement à l'établissement de la session.

Destination ID : 8 bits, identifie le récepteur du paquet. Il est aussi généré aléatoirement à L'établissement de la session.

Session ID: 16 bits, permet d'identifier une session MRTP. Cet identificateur est porté par tous les paquets appartenant à cette session.

Flow ID: 16 bits, ce champ permet l'identification du flux dans une session MRTP. Il est généré aléatoirement à l'établissement de la session ou lorsqu'un nouveau flux vient de joindre la session.

Flow Sequence Number : 32 bits, donne le numéro de séquence d'un paquet dans le flux pour lequel il appartient. Le numéro de séquence initial est généré aléatoirement à l'établissement de la session.

Timestamp : 32 bits, indique l'instant de génération du premier octet de la charge utile.

Payload : représente les données multimédias portées dans le paquet.

Padding : 0 à 3 octets, il est employé pour aligner la frontière de paquet avec un mot de 32 bits.

4.9.2 Les rapports de contrôle de QoS (MRTCP QoS Reports)

On distingue deux types de paquets de contrôle de qualité de service :

- Les paquets envoyés par l'émetteur (Sender Report) : le format d'un paquet SR est montré dans la figure 4.4.
- Les paquets envoyés par le récepteur (Receiver Report) : le format de ces paquets est semblable à celui de SR (Sender Report), avec la suppression des champs : nombre total des paquets et le nombre total des octets.

L'en-tête du paquet SR (Sender report) est semblable à l'en-tête de paquet de données, avec les nouveaux champs suivants :

This FlowID:16bits, représente l'identificateur du flux dans lequel le rapport est envoyé (identifie le flux qui porte le rapport de QoS).

Length : 16 bits, indique la longueur totale du rapport de QoS (en octets).

NTP Timestamp : 64 bits, il indique le moment de wallclock (Wallclock : est l'horloge utilisée pour synchroniser les différents nœuds du réseau). Quand ce rapport est envoyé, il peut être employé en combinaison avec des horodateurs retournés dans les rapports des récepteurs pour mesurer le temps d'aller-retour (RTT).

MRTP Timestamp : 2 bits, c'est l'horodateur de MRTP correspondant au même temps à l'horodateur de NTP. Il peut être employé dans la synchronisation, pour estimer la fréquence de l'horloge de base de MRTP et dans l'évaluation de RTT (le temps d'aller-retour).

Total packet count : 32 bits, le nombre total des paquets de données MRTP envoyés au dessus du même flux. Ce paramètre peut être utilisé par le récepteur pour estimer le nombre de paquets perdus.

Total number of octets: 32 bits, le nombre total des octets de données envoyés au-dessus du même flux.

Cumulative Number of Packets Lost in Flow : 24 bits, le nombre total des paquets perdus dans un flux.

Ver (4bits)	Rved (2bits)	Ext (1bit)	Rved (1bit)	Payload Type (8bits)	Source ID (8bits)	Destination ID (8bits)
Source ID (16bits)				Number of Flow (16bits)		
This flow ID (16 bits)				Length (16 bits)		
NTP Timestamps most significant word (32 bits)						
NTP Timestamps least significant word (32 bits)						
MRTP Timestamp (32 bits)						
Flow ID (16 bits)			Source ID (8 bits)		Destnation ID (8 bits)	
Total packet count (32 bits)						
Total octet count (32 bits)						
Fraction lost in flow (8bits)	Cumulative number of packet lost in flow(24 bits)					
Interarrival jitter (32 bits)						
Highest sequence number received (32 bits)						
Last report received (32 bits)						
Delay since last report (32 bits)						
.....						
Profile specific extensions						

Figure 4.4 : Format de paquet emetteur de MRTP[55]

Interarrival Jitter : 32 bits, c'est le temps entre les arrivées successives des paquets d'un flux.

Highest Sequence Number Received: 32 bits, c'est le numéro de séquence le plus élevé qui a été reçu pour un flux (pour le flux considéré).

Last Report Received: 32 bits, nous employons les bits du milieu, 32 sur 64 dans l'horodateur de NTP du rapport de MRTCP le plus récemment reçu de ce flux.

Delay Since Last Report Received: 32 bits, c'est le temps entre la réception du dernier SR ou RR et l'envoi de ce rapport.

Pour augmenter la fiabilité des rétroactions (feedback), les rapports RRs ou SRs peuvent être envoyés au-dessus des chemins multiples. Dans ce dernier cas, le champ horodateur de MRTP est employé pour examiner les anciens rapports ou les rapports dupliqués. Dans MRTP, l'expéditeur et le récepteur estiment le RTT. Le RTT estimé peut être employé par l'expéditeur pour s'adapter aux conditions de transmission.

4.9.3 Les Messages de contrôle de session / flux

Les messages de contrôles de MRTCP incluent les messages employés pour établir la session MRTP, les messages pour gérer et contrôler l'ensemble des flux utilisés et pour décrire les différents participants de la session MRTP.

4.9.3.1 Message d'établissement et de contrôle de la session :

Il y a trois messages pour l'établissement et le contrôle de la session.

Le message Hello: ce message est envoyé par l'émetteur ou le récepteur (l'initiateur de la session) pour initialiser une session MRTP. Le format de ce message est illustré dans la figure 4.5. Après l'en-tête commun de tous les paquets MRTP, on trouve l'identificateur de la session (généré d'une manière aléatoire) ensuite, le nombre total des flux supportés par la session.

<table>
<tr><td>Ver
(4bits)</td><td>Rved
(2bits)</td><td>Ext
(1bit)</td><td>Rved
(1bit)</td><td>Payload Type
(8bits)</td><td>Source ID
(8bits)</td><td>Destination ID
(8bits)</td></tr>
<tr><td colspan="4">Source ID (16bits)</td><td colspan="3">Number of Flow (16bits)</td></tr>
<tr><td colspan="3">Flow ID (16 bits)</td><td colspan="2">Source ID (8 bits)</td><td colspan="2">Destination ID (8 bits)</td></tr>
<tr><td colspan="7">Source IP adress (32 bits)</td></tr>
<tr><td colspan="7">Destination IP adress (32 bits)</td></tr>
<tr><td colspan="4">Source port number (16 bits)</td><td colspan="3">Destination port number (16 bits)</td></tr>
<tr><td colspan="7">Initial flow sequence number (32 bits)</td></tr>
<tr><td colspan="7"></td></tr>
<tr><td colspan="7">Profile specific extensions</td></tr>
</table>

Figure 4.5 : Format d'un paquet hello session [55]

Le message ACK Hello:est envoyé par le nœud récepteur pour affirmer la réception d'un message hello. Le format de ce message est semblable à celui du message hello avec deux principales différences :

- Le champ type de la charge utile à une valeur différente.
- Le champ qui représente le numéro de séquence initiale du flux est remplacé par le champ : statut du flux. Une valeur de succès pour ce champ indique que le flux envoyé a été confirmé par le noeud destinataire, alors qu'une valeur d'échec indique que le flux est nié pour être employé.

Les messages MRTP Bye session et ACK Bye session: ces messages sont utilisés pour terminer une session MRTP. Le format de message Bye session est donné dans la figure 4.6. Le format de message ACK Bye session est similaire à celui de MRTP Bye session à l'addition du champ statut.

Une valeur de succès signifie que la session MRTP est terminée avec succès et les ressources allouées à cette session sont toutes libérées, alors qu'une valeur d'échec, (ou un arrêt) signifie que la session MRTP est terminée avec échec. Dans ce cas-là, le noeud peut retransmettre le message MRTP Bye session jusqu'à ce qu'un nombre maximum de fois soit atteint.

Ver (4)	Rved (2)	Ext (1)	Rved(1) (1)	Payload type (8)	Source Id (8)	Destination Id (8)
Id session (16)					Number of flow (16)	

Figure 4.6 : Format de message Bye session.[55]

4.9.3.2 Message de contrôle du flux

Pendant la phase de transmission, des flux peuvent être rompus ou encombrés, et d'autres nouveaux flux peuvent joindre la session. Les messages de contrôle du flux nommés Add/Delete Flow et ACKAdd/Delete Flow sont utilisés pour ajouter ou supprimer des flux à partir d'une session MRTP. Les formats des messages Add/Delete Flow et ACKAdd/Delete Flow n est similaire à celui de message hello session et ACK hello session, mais avec une valeur de champ : type de charge utile : (payload type) différent et sans le champ : numéro de séquence initiale du flux (initial flow sequence number).

Description du participant

Comme dans RTP, MRTP emploie la description de la source pour identifier la source et le CNAME pour identifier le récepteur. Dans MRTP, chaque participant est également identifié par un identificateur unique (ID). Les identificateurs (IDS) peuvent être assignés d'une manière aléatoire, ou calculés à partir du CNAME en utilisant une fonction de hachage par exemple.

4.9.4 Extension de l'en-tête

MRTP emploie des en-têtes d'extension pour soutenir des fonctions additionnelles non soutenues par les en-têtes principales. Le format commun des en-têtes d'extension est donné dans la figure 4.7. Le champ type est défini par des profils de MRTP. Les en-têtes d'extension de MRTP ont des longueurs variables. La longueur de chaque extension est indiquée par le champ longueur (length).

.

Ver(4)	Rved(2)	Ext(1)	Rved(1)	Type(8)	Length(16)
		Extention header specific data(variable length)			

Figure 4.7 : Format d'en-tête d'extension [55]

Il existe plusieurs en-têtes d'extension de MRTP. Parmi ces extensions, on cite les suivantes :

Source routing extension header

Puisque le protocole MRTP utilise des chemins multiples alors le routage par source peut être employé pour indiquer explicitement l'itinéraire pour chaque paquet. Cependant, si les couches inférieures ne supportent pas le routage par source, alors le routage par source peut être implémenté au niveau application en définissant une extension de l'en-tête qui indique l'itinéraire pour le paquet.

Authentification extension header

Cet en-tête fournit un mécanisme simple d'authentification en utilisant un champ d'identification et un champ de mot de passe chiffré avec des applications spécifiques de chiffrement. Cette extension d'en-tête peut être employée dans les paquets de contrôle de session/flux pour valider les opérations demandées, ou dans un RR ou un SR pour valider le rapport de QoS.

4.10 Les opérations MRTP/MRTCP

Le schéma 4.8 illustre les différentes opérations d'une session de MRTP

Figure 4.8 : les differents opération MRTP [55]

4.10.1 Etablissement et fermeture d'une session MRTP

MRTP est un protocole orienté connexion d'une façon qu'une session MRTP doit être initialisée avant le commencement du transfert des données. L'initialisation de la session

peut être effectuée soit par l'émetteur ou par le récepteur. Cette opération est constituée de trois messages. Premièrement l'initiateur de la session envoie un message 'Hello session', puis le second côté répond par un message 'ACK Hello session', ensuite le nœud initiateur répond par le message 'ACK Hello session'. Ces trois étapes permettent aux deux extrémités d'éviter des collisions produites pendant la génération aléatoire des identificateurs des flux et des sessions.

4.10.2 Transfert des données

Une fois que la session est établie, des paquets MRTP portant des données multimédias seront transmis dans des flux multiples liés à la session. Chaque paquet a son numéro de séquence qui est local à son flux et un horodateur (timestamp) qui sera utilisé par le récepteur pour ordonner les paquets et synchroniser les flux.

4.10.3 Rapport de qualité de service

Pendant une session MRTP, le récepteur continue la gestion et la surveillance de QoS en envoyant des statistiques sur le nombre des paquets perdus, le plus grand numéro de séquence reçu et la gigue pour chaque flux et session. Ces informations sont encapsulées dans un seul paquet puis ce dernier est envoyé sous forme d'un rapport vers l'émetteur. La fréquence d'envoi des rapports est fixée par l'application.

4.10.4 Gestion des flux

Pendant la session MRTP, il y a des flux qui sont indispensables. Par exemple un nœud intermédiaire est crashé ou encombré, ainsi dans les réseaux Ad hoc un nœud peut quitter le réseau, dans ce cas chacun, soit le récepteur ou l'émetteur peut supprimer le flux en envoyant un paquet de suppression du flux 'delete flow' qui contient l'identificateur de flux à supprimer. Quand une nouvelle route est trouvée, un nouveau flux peut être ajouté en envoyant un paquet 'Add flow packet '.Ces mécanismes permettent au protocole MRTP de réagir rapidement aux changements de topologie et aux congestions du réseau.

4.11 Conclusion

Dans ce chapitre nous avons présenté le protocole MRTP/MRTCP qui englobe le routage multichemins et le mécanisme de contrôle de QoS (feedback). Ces deux mécanismes utilisés par ce protocole permettent à l'émetteur de s'adapter aux conditions de la transmission. Le routage multichemins permet de transmettre les paquets d'un flux en utilisant des chemins qui sont partiellement ou totalement disjoints[(1)]. Ceci permet d'équilibrer

[(1)] Deux chemins sont dits totalement disjoints s'ils ne comportent aucun lien commun et ils sont partiellement disjoints s'ils contiennent au moins un lien commun.

la charge et de diminuer le nombre des paquets perdus, car si un chemin est craché seulement les paquets traversant ce chemin seront perdus.

Malgré les avantages introduits par ce protocole, ce dernier ne protège pas les premiers flux. Autrement dit, les premiers flux (les flux envoyés juste après l'établissement de la session) peuvent souffrir d'un délai de bout en bout supérieur à celui exigé par l'application. Cet inconvénient est très grave pour les applications sensibles au délai tel que la voix.

Dans le prochain chapitre, nous présenterons notre proposition appelée : MRTP avec contrôle d'admission qui permet de surpasser ce problème.

5

MRTP avec contrôle d'admission

5.1 Introduction

Le protocole MRTP/MRTCP est un protocole qui permet le transport des données multimédias dans les réseaux Ad hoc. Il utilise des chemins multiples partiellement ou totalement disjoints. Le transfert des données à travers des chemins multiples permet de diminuer le nombre des paquets perdu, car au cas où un chemin est rompu seulement les paquets envoyés à travers ce chemin sont perdus. D'autre part, les nœuds récepteurs dans MRTP envoient des rapports qui contiennent des statistiques sur les paramètres de QoS. Ces rapports permettent à l'émetteur de s'adapter à la situation du réseau. Mais ce protocole ne permet pas de choisir des chemins qui satisfont le délai de bout en bout exigé par une application. Donc si par exemple les chemins choisis ont un délai de bout en bout >200ms et le trafic envoyé est de type voix, alors les paquets envoyés à travers ces chemins n'ont aucune signification et seront considérés comme des paquets perdus.

D'autre part, le modèle SWAN, que nous avons présenté dans le troisième chapitre, propose une solution pour ce problème. SWAN utilise un contrôle d'admission sur le flux à temps réel. C'est-à-dire si le délai de bout en bout est inférieur au délai exigé par l'application alors le flux est accepté sinon le flux est rejeté. Mais l'inconvénient de cette méthode est l'utilisation d'un seul chemin, en cas de rupture de chemin, tous les paquets envoyés, seront perdus. Ainsi, l'utilisation d'un seul chemin provoque la congestion des nœuds si le réseau est chargé.

Pour pallier à ces problèmes, nous avons développé une méthode qui combine les avantages de deux méthodes précédentes (MRTP + SWAN). Cette méthode est appelée MRTP avec contrôle d'admission. Dans ce qui suit, nous présentons les détails de cette méthode.

5.2 Proposition

Notre proposition consiste à utiliser le protocole MRTP/MRTCP et à ajouter les modules suivants afin de trouver une solution aux problèmes cités précédemment.

- **Le module Classificateur:** permet de classifier les flux entrant à un nœud en différenciant entre les flux « best effort» et les flux de type voix.
- **Le module contrôle d'admission :** permet l'admission ou non d'un flux voix en se basant sur le délai de bout en bout.
- **Le module mesure de QoS :** calcule le délai de bout en bout, de chaque chemin ensuite envoie, ces délais vers le module contrôle d'admission.

Le schéma de notre proposition est donné dans la figure 5.1

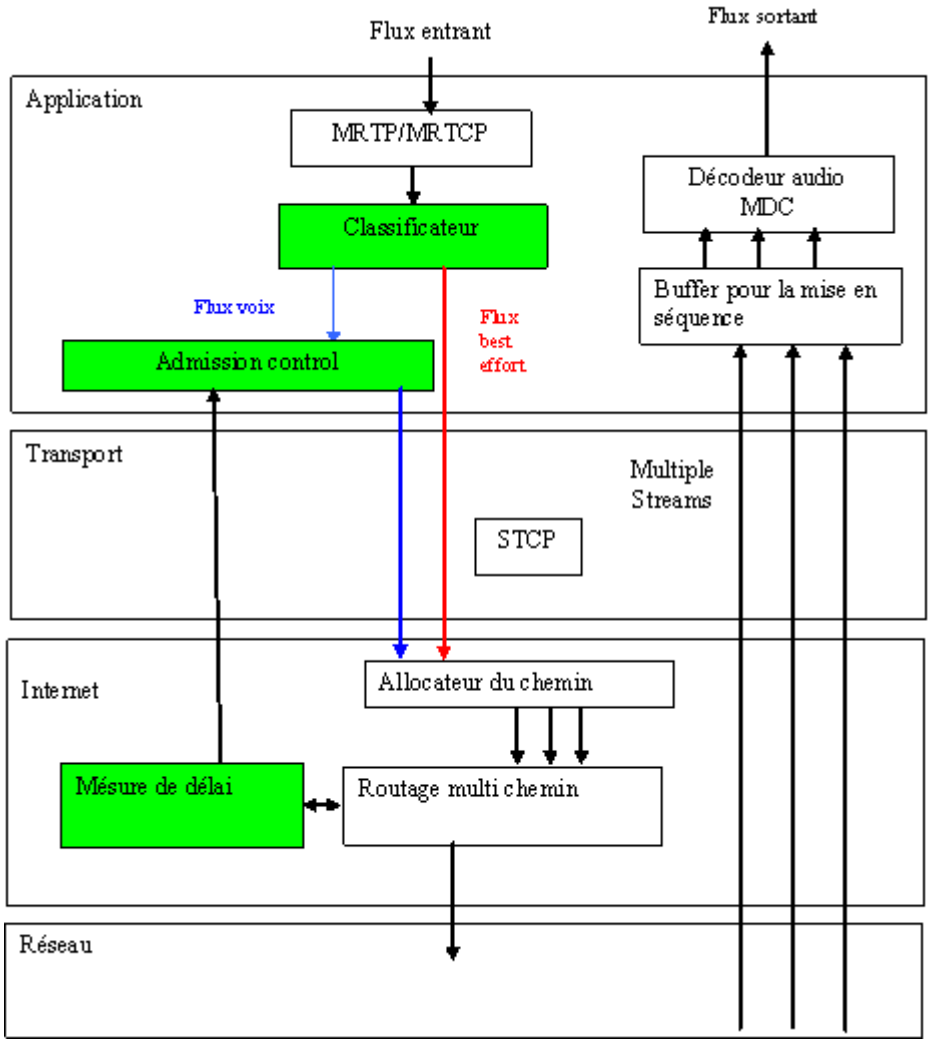

Figure 5.1 : MRTP avec contrôle d'admission

5.2.1 Module classificateur

Il permet de classifier les paquets arrivant à deux catégories :

Les paquets de type voix : ce flux est envoyé vers le module de contrôle d'admission. Au niveau du module contrôle d'admission le flux est accepté s'il existe des chemins entre la source et la destination qui satisfont le délai exigé par la voix (<=200ms). Si aucun chemin ne satisfait le délai exigé par la voix, le module contrôle d'admission génère un message d'erreur : « flux non accepté ».

Les paquets de type Best effort (les données qui ne sont pas de type voix) : ces paquets seront orientés directement vers le module allocateur de chemin, afin d'être transmis vers la destination.

Pour classifier les différents paquets, le classificateur se base sur le champ : charge utile (playload),qui se trouve au niveau de l'en-tête de paquet MRTP selon les codes indiqués dans la table 5.1.donc si le champ: charge utile est egal un valeur comme(0,1,2,3,4....)alors le paquet et de type audio, si le champ est egal à une valeur qui indique la video comme(24, 31, 34...) alors le paquet est de type video.

Code	Type d'information	Code	Type d'information
0	PCMU audio	15	G.728 audio
1	1060 audio	16-22	audio
2	G.721 audio	23	RGBB video
3	GSM audio	24	HDCC video
4	audio	25	Celb vedeo
5	DV14 audio(8khz)	26	JPEG video
6	DV14 audio(16khz)	27	CUSM video
7	LPC audio	28	nv vedeo
8	PCMA audio	29	PicW video
9	G 722 audio	30	CPV video
10	L16 audio (stéreo)	31	H.261 video
11	L16 audio (mono)	32	MPV video
12	LPS0 audio	33	MP2T video
13	VSC audio	34	H.263 video
14	MPA audio		

Table 5.1: les valeurs possibles pour le champs payload de MRTP

A la réception d'un paquet par le classificateur, ce dernier exécute le programme suivant :

Algorithme 5.1 *: classificateur ()*

1:**Var** charge : entier
2:Debut
3: Copier (charge, paquet, payload)
4: **Si** charge = voix **alors**
5: Envoyer paquet vers le module de contrôle d'admission
6: **Sinon**
7: Envoyer paquet vers le module « allocateur de chemins »
8: **FSI**
9:FDebut

5.2.2 Contrôle d'admission

Le module contrôle d'admission décide l'acceptation ou le rejet d'un flux voix en se basant sur les délais de bout en bout des chemins liant la source S et la destination D. c'est-à-dire s'il y a des chemins entre la source et la destination qui satisfont le délai exigé par la voix (délai <=200ms) alors le flux est accepté. Sinon ce module génère un message d'erreur : « flux non accepté ».

Pour écrire l'algorithme de ce module, nous utilisons les variables et les fonctions suivantes :

Nombre_chemin : le nombre de chemins existant entre la source et la destination.

Liste_chemins : la liste de différents chemins liant la source et la destination. Chaque maillon de la liste représente un chemin. Il est composé de deux champs :

Champ1 : contient les adresses des nœuds à traverser pour atteindre la destination.

Champ2 : contient le délai de bout en bout associé à ce chemin.

Liste_valide : une liste qui contient les différents chemins satisfaisant le délai de bout en bout exigé par l'application voix.

Nombre_valide : le nombre des chemins valides.

Envoyer_flux (chemin_valide, flux_voix) **:**permet d'envoyer le flux voix à travers la liste des chemins valides.

Inserer (chemin_i, liste1): permet d'insérer le chemin i dans la liste nommée liste1.

A la réception d'un flux voix, le module de contrôle d'admission exécute l'algorithme suivant:

Algorithme 5.2 *Algorithme Admission_ control()*

```
1:Var i, Nombre_chemin : entier ;
2:      liste_chemin, liste_valide : liste ;
3:      liste_valide= vide ;
4:      Nombre_valide=0 ;
5:Debut
6:    Pour i=1 à Nombre_chemin faire
7:      Debut
8:         Si liste_chemin[i].champ2<=200 alors /*On teste s le délai de
                                                  chemin i est inférieur à 200ms */
9:            chemin_i = liste_chemin[i].champ1
10:             Inserer( chemin_i, liste_valide) /* On insère le chemin i dans
                                                  la liste des chemins valides */
11:             Nombre_valide= Nombre_valide +1 /* On incrémente le nombre
                                                  des chemins valides par 1*/
12:        Fsi
13:    Fpour
14: Si  (Nombre_valide >=2)alors
15:   Envoyer_flux (chemin_valide, flux_voix)
16: Sinon
17:    Generer_message ( 'flux non accepté')
18: FSI
19:Fdebut
```

5.2.3 Le module mesure du delai

Le module « mesure du délai » permet de mesurer le délai de bout en bout de chaque chemin entre la source et la destination.

L'émetteur génère un paquet dont la taille est semblable à la taille d'un paquet voix, ensuite il diffuse ce paquet à travers tous les chemins liant la source et la destination. Le paquet généré contient les informations suivantes :

- Adresse du nœud source (ID_source).
- Adresse du nœud destinataire (Id_destination).
- Le chemin à traverser pour atteindre la destination (chemin_i).
- Un champ qui indique le délai (délai).

Pour calculer le délai de bout en bout, nous utilisons deux méthodes selon le type du réseau.

Réseau symétrique

Si le réseau est symétrique, c'est a dire si la source A peut atteindre la destination D en utilisant le chemin ABCD alors la destination peut atteindre la source en utilisant le chemin inverse DCBA. Donc si le réseau est symétrique, le nœud source effectue les taches suivantes :

- Génère un paquet.
- Déclenche un timer.
- Initialise le champ délai de paquet généré par la valeur du timer.
- Diffuse un paquet qui contient les informations décrites précédemment à travers tous les chemins.

A la réception de chaque paquet par la destination, la destination inverse le chemin (chemin_i) porté par le paquet, ensuite renvoi ce paquet vers le nœud source. Quand la source reçoit le paquet envoyé par la destination, elle calcule le délai d'aller-retour (RTT) comme suite :

$RTT = timer - délai$.....(1)

Ensuite le délai de bout en bout D est donné par :

$\boxed{D = \frac{RTT}{2}}$.....(2)

Ensuite, ce délai est enregistré avec le chemin dans la liste nommée: **liste_chemin.**

Donc, le module de mesure de QoS s'exécute au niveau de l'émetteur et au niveau du récepteur. Pour écrire l'algorithme exécuté par ce module, nous définissons les variables suivantes :

Liste_chemin : une liste dans laquelle on sauvegarde tous les chemins liant la source et la destination avec le délai de chaque chemin. Elle est composée de deux champs :

Champ1 : contient les adresses de différents nœuds qui permettent d'atteindre la destination à partir de la source.

Champ2 : contient le délai de bout en bout associé à ce chemin.

RecuD : une table de booléens qui indique pour la source la réception ou non d'un paquet à partir de la destination sur le chemin i. si reçu [i] =1 c'est-à-dire que le nœud source a reçu un paquet sur le chemin i.

RecuS : une table de booléens qui indique pour la destination, la réception ou non d'un paquet à partir de la source sur le chemin i. si reçu[i] =1 c'est-à-dire que le nœud destination a reçu un paquet sur le chemin i.

Nombre_chemin : le nombre des chemins existant entre la source et la destination.

Envoyer_paquet(p, id_source, id_destination, chemin_i): permet d'envoyer un paquet p à partir du nœud d'adresse Id_source vers le nœud d'adresse Id_destination sur le chemin_i.

Copier (liste_chemin, chemin_i, $\mathbf{D}_{sd}$):permet de copier le chemin_i avec son delai $\mathbf{D}_{sd}$ dans la liste nommée **liste_chemin.**

Au niveau du récepteur:

Algorithme 5.3 *Algorithme QoS_recepteur()*

```
Var i : entier ;
 Chemin_i : liste ;
  RecuS : tableau boolean;
  Debut
    Pour i=1 a Nombre_chemin faire
       Si recuS[i] /*si un paquet est reçu sur le chemin i */
          Inverser_chemin(chemin_i)
          Envoyer_paquet(p, id_destination, id_source, chemin_i)
         Fsi
     Fpour
    Fdebut
```

Au niveau de émetteur

Algorithme 5.4 *Algorithme QoS_ emetteur()*

Var i,j :entier
Liste_chemin :liste /* chaque maillon de la liste pointe vers une liste des adresses menant de la source vers la destination*/
Chemin_i : tableau d'entiers
RecuD : tableau de boolean
Debut
generer_paquet(p)
p.delai = Timer()
Pour i=1 a Nombre_chemin **faire**
Envoyer_paquet(p, id_source, id_destination, chemin_i)
Fpour
J= Nombre_chemin
Tanque (j>0) **faire**
Debut
Si recuD[j] **alors** /* si un paquet est reçu sur le chemin i a partir de la destination*/
$D_{sd} = \frac{timer - p.délai}{2}$
Copier (liste_chemin, chemin_j, D_{sd})
j = j-1
Fsi
FTQ
Envoyer **liste_chemin** vers le module controle d'admission
FDebut

Réseau asymétrique (non symétrique)

Si le réseau est asymétrique alors nous ne pouvons pas calculer le délai exact. Donc nous supposons la synchronisation des nœuds, ensuite nous estimons le délai de bout en bout en utilisant une méthode d'estimation de délai. Plusieurs méthodes ont été développées. Par exemple [57], [58], [59], [60], [61], [62], [63]. mais la plupart de ces travaux essayent d'estimer le délai en se basant sur la modélisation de la DCF de 802.11 faite par Bianchi, ce dernier, estime en priorité les probabilités de collision et d'accès au canal radio. Cependant, les hypothèses mises dans le modèle de Bianchi, ne sont pas réalistes et ne peuvent être

adaptées pour le calcul du délai dans un environnement Ad hoc multisauts. Ces hypothèses sont :

- Le nombre n de stations en compétition est connu et constant.
- La probabilité de collision est constante et indépendante.
- Cette probabilité de collision est la même pour tous les nœuds.
- Les noeuds sont tous dans la même zone de communication.

Toutes ces hypothèses fondamentales permettent de simplifier les chaînes de Markov obtenues et de les résoudre, ce qui ne sera pas le cas dans des réseaux multisauts.

Nous utilisons ici une méthode qui tire la valeur de la probabilité de collision à partir de simulation.

Détermination du délai à un saut radio Le délai de transmission à un saut radio peut se décomposer en deux parties :

– Le délai entre l'instant où le paquet entre dans la file d'attente du noeud émetteur et l'instant où il est passé à la couche MAC.

– Le délai s'écoulant entre le moment où le paquet est reçu par la couche MAC jusqu'à la réception de l'acquittement correspondant par le noeud récepteur.

Un paquet se trouvant à une station quelconque provient de deux sources : les paquets générés localement au niveau du noeud considéré et les paquets routés qui passent par ce noeud. Ainsi, chaque noeud du réseau peut jouer le rôle de noeud source, relais ou destination.

Détermination du délai dans la file d'attente Un noeud sans fil 802.11 peut être vu comme un buffer qui se remplit par des paquets entrants provenant des couches supérieures. Ainsi, un seul serveur fournit le traitement nécessaire pour ces paquets. Nous pouvons donc modéliser ce système par une file d'attente M/M/1/K (voir figure 5.2) possédant les propriétés suivantes :

– La distribution du temps inter-arrivée des clients est exponentielle de paramètre λ

– Le traitement des clients suit également une loi exponentielle de paramètreμ

– Il y a un seul serveur pour le traitement des clients entrants

– La taille de la file est bornée par la valeur K, lorsqu'un client arrive et qu'il y a déjà K clients dans le système alors celui-ci est perdu.

Le paramètre λ représente le débit désiré par l'application qui est explicitement fourni lors de la phase de requête de route avec QoS. Le paramètre μ représente la bande passante libre autour de ce mobile. Cette valeur peut être estimée en calculant le pourcentage de temps libre qui sera ensuite multiplié par la capacité du médium radio.

donc le délai dans la file est donné par:

$$R = \frac{\rho}{1-\rho} * \frac{1-(k+1)\rho^k + k\rho^{k+1}}{1-\rho^k} * \frac{1}{\lambda} \; si \; \rho \neq 1 .....(3)$$

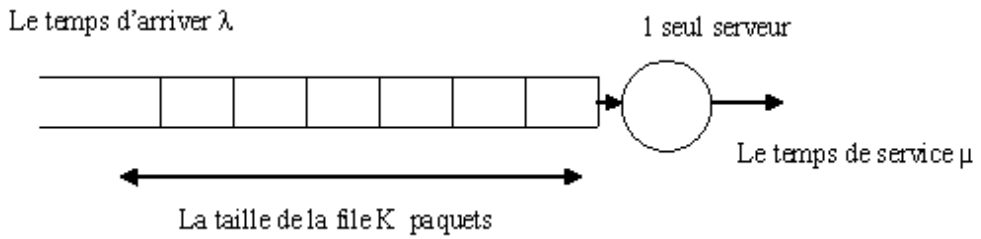

Figure 5.2 : Système avec file d'attente à K paquets

$R = \frac{1}{2}(k+1) * \frac{1}{\lambda} \qquad si\ \rho = 1$.....(4)

avec $\rho = \frac{\lambda}{\mu}$.....(5)

Détermination du délai de transmission

Le délai de transmission est le délai qui s'écoule depuis l'arrivée d'un paquet à la couche MAC jusqu'à la réception du paquet d'acquittement en provenance du nœud récepteur, incluant toutes les retransmissions en cas de collision.

Soit p la probabilité de collision sur le lien considéré, n le nombre de retransmissions associées à la probabilité de collision p et X la variable aléatoire représentant le nombre de retransmissions. On a donc les égalités suivantes en terme de probabilité.

$p(X = 0) = 1 - p$.....(6)

$p(X = 1) = p(1 - p)$.....(7)

$p(X = 2) = p^2(1 - p)$.....(8)

.

.

$p(X = k) = p^k(1 - p).\ si\ k <= 6$.....(9)

Dans la norme IEEE 802.11 le nombre maximum de retransmissions est fixé à 7 donc:

$p(X = 7) = p^7\ si\ k = 7$.....(10)

$p(X = k) = 0\ si\ k >= 8$.....(11)

Nous pouvons donc en déduire l'espérance de la variable aléatoire X correspondant au nombre moyen de retransmissions n.

$n = E(x) = \sum_{k=0}^{6} k * p^k(1 - p) + 7p^7$.....(12)

après calcul on trouve :

$n = \frac{6p^8 - 7p^7 + p}{1 - p}$.....(13)

Selon la norme IEEE 802.11, à chaque collision la taille de la fenêtre de contention est doublée(on utilise l'algorithme: binary exponential backoff)donc la taille de la fenêtre de backoff CW est une suite croissante d'entiers de la forme $2^k - 1$, ainsi à la k^{eme} collision successive (k <= 7 selon la norme IEEE 802.11) la taille de la fenêtre de backoff est de $2kCWmin$. Le backoff suit une loi uniforme à chaque stage de backoff, nous prendrons la valeur moyenne pour représenter la valeur du backoff qui sera choisi pour le paquet qui va

être émis. Ainsi le délai de propagation sur le canal noté D_{prop} est donné par la formule suivante:

$D_{prop} = \sum_{k=0}^{n}(DIFS + \frac{1}{2} * 2^k W_{\min} + T_m)$.....(14)

Aprés calcule on trouve:

$D_{prop} = (n+1)(DIFS + T_m) + \frac{1}{2}W_{\min}(2^{n+1} - 1)$.....(15).

$W_{\min}$:la taille initiale de la fenêtre de backoff.

T_m :représente le temps de propagation d'un paquet de taille m. ce délai est donné par la formule:

$$T_m = \frac{la\ longueur\ de\ paquet}{le\ debit\ de\ transmission}$$.....(16)

$DIFS$:le délai que doit attendre un noeud avant la transmission pour assurer que le canal est libre (voir chapitre2, mode accès DCF).

Donc le délai d'un paquet sur un lien entre deux noeuds est donné par:

$D = R + (n+1)(DIFS + T_m) + \frac{1}{2}W_{\min}(2^{n+1} - 1)$ .....(17)

Détermination du délai multi sauts Le délai moyen de bout en bout entre une source s et une destination d est égal à la somme des délais moyens des liens constituants ce chemin. Donc le délai de bout en bout est donné par:

$\boxed{delai = \sum_{i \in s,d} D_i}$.....(18)

Avec D_i :le délai de chemin i calculé par la formule 17.

Si le réseau est asymétrique alors le délai est estimé par le noeud destination en utilisant la formule (18). Ensuite ce délai est inséré dans un paquet. Ce paquet sera envoyé vers le noeud source. A la réception de ce paquet par le noeud source, ce dernier insère le délai et le chemin dans la liste nommée **liste_chemin.**

5.2.4 Relation entre le protocole MRTP/MRTCP et le module contrôle d'admission

Après le calcul de la liste des chemins valides par le module de contrôle d'admission (c'est-à-dire la liste des chemins satisfaisant le délai de bout en bout exigé par l'application voix), ce dernier communique cette liste au module MRTP/MRTCP. Le module MRTP/MRTCP initialise une session MRTP en envoyant la liste des chemins valides vers la destination, pour que les deux côtés soient d'accord sur les chemins à utiliser. Ensuite le flux est envoyé vers le module « classificateur» ce dernier distingue entre le flux voix et autres flux en se basant sur la charge utile (payload) portée dans l'en-tête des paquets MRTP. Puis les paquets voix seront orientés vers le module contrôle d'admission. A l'arrivée de ces paquets au niveau de ce module s'il y a des chemins qui satisfont le délai de bout en bout exigé par la voix, le flux est accepté et les paquets seront envoyés, vers le module allocateur de chemins afin d'être dispersé et envoyé à travers les différents chemins validés par le module « contrôle d'admission ». Mais s'il n'existe pas des chemins qui satisfont le délai de bout en bout

exigé par la voix, même si MRTP a initié une session avec la destination, le module contrôle d'admission refuse le flux voix et génère un message « flux non accepté ».

5.2.5 Exemple de déroulement de la proposition

Soit le réseau donné par la figure 5.3. Le nœud numéro 1 est le nœud source et le nœud numéro 9 est le nœud destination.

Nous supposons que le réseau est symétrique.

Nous notons par :

$RTT1$: le délai d'aller-retour du premier chemin (1, 2, 3, 4,9) et $d1$ le délai de bout en bout du même chemin.

$RTT2$: le délai d'aller-retour du deuxième chemin (1, 5, 6,9) et $d2$ le délai de bout en bout du même chemin.

$RTT3$: le délai d'aller-retour du troisième chemin (1, 7, 8,9) et $d3$ le délai de bout en bout du même chemin. Ipn : l'adresse IP du nœud n.

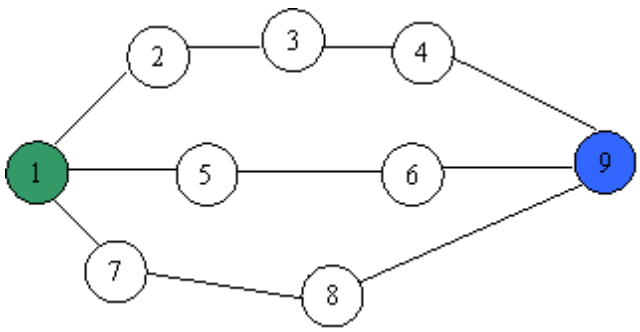

Figure 5.3 : Exemple d'un reseau Ad hoc

Le nœud numéro 1 envoie trois paquets à travers les trois chemins. Ces paquets contiennent les informations suivantes :

Le contenu du paquet envoyé sur le chemin numéro1 est donné par la table 5.2

Id_source	Id_destination	chemin_i	delai
Ipde noeud1	Ipde noeud9	Ip1,Ip2,Ip3,Ip4,Ip9	timer()

Table 5.2: Constituants du paquet de chemin1

Le contenu du paquet envoyé sur le chemin numéro2 est donné par la table 5.3

Id_source	Id_destination	chemin_i	delai
Ip de noeud1	de noeud9	Ip1,Ip5,Ip6,Ip9	timer()

Table 5.3: Constituants du paquet de chemin2

Le contenu du paquet envoyé sur le chemin numéro3 est donné par la table 5.4.

Id_source	Id_destination	chemin_i	delai
Ipde noeud1	Ipde noeud9	Ip1,Ip7,Ip8,Ip9	timer()

Table 5.4: Constituants du paquet de chemin3

Aprés l'allez vers la destination et le retour à la source de chaque paquet, le noeud source calcule le délai d'aller-retour. Supposons que les résultats du calcul sont les suivants :

$RTT1 = 600ms$ donc $d1 = 600/2 = 300ms$

$RTT2 = 300ms$ donc $d2 = 300/2 = 150ms$

$RTT3 = 340ms$ donc $d3 = 340/2 = 170ms$

Champ1	Champ2
Ip1,Ip7,Ip8,Ip9	300ms
Ip1,Ip5,Ip6,Ip9	150ms
Ip1,Ip7,Ip8,Ip9	170ms

Table 5.5: Le de contenue de liste_chemin

Aprés ces calculs, le module de mesure du délai insère ces informations dans la liste nommée : liste_chemin. Donc, liste_chemin contient les informations données dans la table 5.5.Ensuite, cette liste est envoyée vers le module contrôle d'admission.

Le module contrôle d'admission compare le délai de chaque chemin avec le délai exigé par la voix 200ms,ensuite, décide la validité, ou la non validité du chemin. Après la comparaison, ce module valide les deux chemins : (1, 5, 6,9) et (1, 7, 8,9).

Puisque dans cet exemple, on a deux chemins valides, à l'arrivée d'un flux voix, le contrôle d'admission accepte ce flux et envoie les deux chemins vers le module MRTP/MRTCP, afin que ce dernier établisse une session entre la source et la destination. Quand la session est établie le flux voix peut être envoyé.A l'arrivé de ces paquets au niveau de contrôle d'admission seront envoyés vers l'allocateur de chemins pour les envoyer à travers les différents chemins.

5.3 Conclusion

Le protocole MRTP/MRTCP a été développé par Shiwen Mao, pour la transmission de la vidéo sur les réseaux Ad hoc. Ce protocole utilise une technique de routage multichemins

ce qui permet de transmettre les paquets d'un flux en utilisant des chemins différents, par conséquent si un chemin est rompu seulement les paquets transmis à travers ce chemin sont perdus. Ainsi, cette technique de routage permet un bon équilibrage de la charge. En plus de cette technique, MRTP utilise des feedbacks (des rapports qui contiennent des statistiques sur la QoS envoyés par la destination vers la source) pour contrôler la QoS d'une manière périodique. Mais ce protocole ne protège pas les paquets des premiers flux envoyés (c'est-à-dire que les paquets des premiers flux peuvent souffrir d'un délai supérieur à celui exigé par l'application, car MRTP ne prend pas en considération la contrainte de délai pour déterminer les chemins).Dans l'architecture « MRTP avec contrôle d'admission » que nous avons proposé nous avons utilisé ce protocole pour la transmission de la voix et nous avons ajouté les trois modules « classificateur », « contrôle d'admission » et « mesure de QoS » afin d'envoyer les paquets voix seulement à travers les chemins ayant un délai inférieur a celui exigé par la voix, ce qui protège ces paquets.

Pour valider notre proposition nous souhaitons la simuler en utilisant un simulateur du réseau NS2.en modifiant la charge du réseau (on considère les deux cas : réseau charge et réseau non charge) et la mobilité des nœuds pour voir l'influence de ces deux paramètres sur le délai de bout en bout, la gigue et le taux de perte. Ensuite comparer les résultats de notre proposition avec les résultats du protocole MRTP/MRTCP, en terme de nombre des paquets perdus, la gigue, le délai de bout en bout.

6

Conclusion Generale et perspectives

On assiste ces dernières années à une importante évolution dans la société de l'information, conduite par la commercialisation et l'émergence des appareils de communications (tels que les téléphones cellulaires, les ordinateurs portables, les assistants personnels, etc.) et la convergence des réseaux fixes et mobiles. L'utilisateur passe ainsi de l'âge de l'ordinateur personnel à l'âge de l'ubiquité du traitement à travers plusieurs infrastructures. Il a accès à l'information n'importe où et n'importe quand.

Un utilisateur mobile peut consulter son courrier électronique, naviguer sur Internet dans les aéroports, les gares ou dans d'autres lieux publics. Dans une conférence, les chercheurs peuvent transférer des fichiers et d'autres types d'information grâce à leurs appareils électroniques via des réseaux locaux sans fil. Actuellement, les applications distribuées manipulent à présent tous les types de médias au premier lieu l'audio et la vidéo. En Comparaison avec les applications classiques, les applications multimédias présentent de nouvelles contraintes sur le transfert de certains de leurs médias (plus spécifiquement l'audio et la vidéo), tel que le délai de bout en bout, la gigue (la variation du temps d'arrivée des paquets), les pertes des paquets et la bande passante.

Le transport et la gestion des flux temps réel (exemple la voix) dans un réseau IP (Internet) pose un certain nombre de problèmes liés aux variations de la gigue, aux rallongements des délais et aux pertes de paquets. Ces problèmes deviennent encore plus graves dans un contexte de réseaux dynamiques, comme dans le cas des réseaux Ad hoc, où ces paramètres de qualité de service (QoS) doivent encore être mieux contrôlés et maîtrisés que dans le cas de l'Internet filaire.

Pour résoudre ces problèmes plusieurs travaux ont été réalisés soit pour les réseaux filaires tels que : DIFFSEV ou INTSERV ou pour les réseaux Ad hoc comme : SWAN, FQMM, INSIGNIA, MRTP, ou pour les réseaux mixtes (filaires + Ad hoc). Ces recherches ont été basées sur l'introduction d'une différenciation de traitement entre les différents flux ou des mécanismes de feedback. D'autres recherches ont été effectuées aussi dans le secteur de cheminement. Ces modèles cherchent des routes qui répondent à certains critères de QoS

et aujourd'hui ces recherches ont conduit à des protocoles qui sont considérés comme assez mûr pour faire face à des contraintes d'énergie et au changement rapide et fréquent de la topologie du réseau provoquée par la mobilité des nœuds.

Dans ce mémoire nous avons étudié la transmission de la voix sur IP dans les réseaux filaires et les réseaux Ad hoc, les différents paramètres affectant la transmission de la voix sur IP. Puis les solutions proposées pour pallier à ces problèmes, soit dans les réseaux filaires ou les réseaux Ad hoc. Ensuite, nous avons proposé une architecture qui permet d'assurer la QoS pour la voix dans les réseaux Ad hoc. Cette architecture est basée sur le routage multichemins et le contrôle d'admission au niveau de la source. Autrement dit, un flux est accepté au niveau de la source s'il y a des chemins qui satisfont le délai exigé par la voix, une fois le flux est accepté les paquets seront dispersés en utilisant les différents chemins. Aussi, des rapports portant des statistiques sur la QoS seront envoyés par la destination vers la source d'une manière périodique.Cette architecture permet d'envoyer les paquets voix seulement à travers les chemins ayant un délai inférieur à celui exigé par la voix et de protéger ces paquets par l'envoi périodique des rapports de QoS.

Comme perspectives de notre travail, nous souhaitons :

- Simuler le protocole MRTP avec contrôle d'admission pour le comparer avec d'autres protocoles et en particulier avec le protocole MRTP
- Etendre ce protocole afin qu'il assure une bonne QoS pour les réseaux mixtes (filaires + Ad hoc).

Bibliographie

[1] I.Busse, B.Deffner, and H.Schulzrinne. Dynamic QoS Control of Multimedia Applications based on RTP. I*n First International Workshop on High Speed Networks and Open Distributed Platforms*, (St.Petersburg, Russia) 30 Mai1995.

[2] J.L. Mélin. *Qualité de service sur IP.* édition Eyrolles, 2001, ISBN 2-212-009261-x.

[3] G.Pujolle. *Initiation aux réseaux cours et exercices.* édition Eyrolles, 2000, ISBN 2-212-09155-9.

[4] P.B.Vellosso,M.G.Rubinstein, and O.C. M.B.Duarte. Analyzing voice transmission capacity on ad hoc networks. *International Conference on Communication Technology Proceedings, ICCT* 2003.

[5] R.Meraihi. *Gestion de la qualité de service et contrôle de topologie dans les réseaux ad hoc.* thèse de doctorat, 2005, Ecole nationale supérieure des télécommunications, Spécialité : Informatique et Réseaux, Paris, France

[6] S. Armenia, L. Galluccio, A. Leonardi,and S. Palazzo. Transmission of VoIP Traffic in Multihop Ad Hoc IEEE 802.11b Networks: Experimental Results.*In Proceedings of the First International Conference on Wireless Internet (WICON'05),*2005.

[7] S.Corson, J. Macker. Mobile Ad hoc Networking (MANET) : Routing Protocol Performance Issues and Evaluation Considerations. Request for Comments 2501,IETF, janvier 1999

[8] Union internationale des télécommunication. Le Rapport essentiel sur la téléphonie IP par le groupe experts de la téléphone IP/ UIT-D. 2003

[9] C. Chassot. *Contribution aux protocoles et aux architectures de communication de bout en bout pour la QoS dans Internet.* Mémoire d'habitation à diriger des recherches. Institut national polytechnique de Toulouse, France, 12 decembre 2005.

[10] G.Auriol. *Spécification et implémentation d'une architecture de signalisation a gestion automatique la QoS dans un environnement IP multi domaines*. Thèse de doctorat institut national des sciences appliquées de Toulouse, Spécialité : Réseaux et Télécommunications, France, 16 novembre 2004.

[11] C. Chaudet. *Autour de la réservation de bande passante dans les réseaux Ad hoc.*Thèse de doctorat, Institut national des sciences appliquées de Lyon, France, 28 septembre 2004.

[12] G.S .Ahn, A.T.Campbell,A.Veres, and L.H.Sun. Supporting Service Differentiation for Real-Time and Best-Effort Traffic in Stateless Wireless Ad hoc Networks (SWAN).*IEEE transaction on mobile computing*, vol 1(3),pages:192- 207, 2002.

[13] S. Corson and J. Macker. Mobile ad hoc networking (manet): Routing protocol performance issues and evaluation considerations. RFC 2501, Janvier 1999.

[14] J. Broch, D. B. Johnson, and D. A. Maltz. The dynamic source routing protocol for mobile ad hoc networks. IETF draft, 16 Avril 2003.

[15] C. E. Perkins, E. M. Belding-Royer, and S. Das. Ad hoc on demand distance vector (aodv) routing. RFC 3561, juin 2003.

[16] C. E. Perkins and E. M. Belding-Royer. Quality of service for ad hoc on-demand distance vector. Internet Draft, 14 Octobre 2003.

[17] L. Leung, R. Jilei, L.Poon, E.Chan, A.-L.C. Baochun. MP-DSR: A QoS aware multipath dynamic source routing protocol for wireless ad hoc networks. *In the Proceedings of the 26th Annual IEEE Conference on Local Computer Networks* (*LCN'01*)Pages:132 - 141, Tampa, Florida, Novembre 2001.

[18] J. Broch, D. A. Maltz, D. B. Johnson, Y. Hu, and J. Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. *In the Proceedings of MobiCom*, Pages 85–97. ACM Press, 1998.

[19] V. D. Park and M. S. Corson. A highly adaptive distributed routing algorithm for mobile wireless networks. *In the Proceeding of IEEE Conference on computer communications* (*INFOCOM'97*), pages 1405–1413, Kobe, Japan, Avril 1997.

[20] J. Xue, P. Stuedi, and G. Alonso. Asap: An adaptive QoS protocol for mobile Ad hoc networks. *In the Proceeding of 14th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications* (*IEEE PIMRC'2003*), Beijing, china, Septembre 2003.

[21] R. Sivakumar, P. S. Inha, and V. Bharghavan. Cedar: Acore-extraction distributed ad hoc routing algorithm. *IEEE Journal on Selected Areas in Communications*, vol :17(8), pages:1454–1465, Aout 1999.

[22] S.D.Bhupau, P. P. Joshi, V.Sahdev, D. D. Callahan « End-to-end Voice over IP Testing and the Effet of QoS On Signaling. IEEE 2003.

[23] "European IP Testbed (Implementation Phase)" Project P803-PF Aout 1999

[24] S.B. Lee, and A.T. Campbell. INSIGNIA: in band signaling support for QoS in mobile Ad hoc networks. *In proceedings of 5th International Workshop on Mobile Multimédia Communication (MoMuc'98)*. 12-14 , Berlin,Octobre 1998.

[25] M.C.Domingo et D.Remondo. A cooperation model between Ad hoc networks and fixed networks for service differentiation. *In Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (LCN'04)*. Pages: 692 – 693. IEEE Computer Society Washington, USA, 2004.

[26] M.C Domingo et D.Remondo.Analysis of VBR Voip Traffic for Ad hoc connectivity with fixed IP network. *Proceedings of IEEE Vehicular Technology Conference(VTC'2004)*, Fall, Los Angless,C.A, U.S.A,2004

[27] J. Rosenberg, et al. SIP: Session Initiation Protocol. IETF RFC 3261, June 2002.

[28] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A transport protocol for realtime applications. IETF Request For Comments 1889. [Online]. Available: http://www.ietf.org.

[29] W. R. Stevens, TCP/IP Illustrated. Volume 1: The Protocols. Reading, MA: Addison-Wesley, 1994.

[30] Y. Nebat and M. Sidi. Resequencing considerations in parallel downloads. *In Proceedings. IEEE. 21th Annual Joint Conference of the IEEE Computer and Communications Societies. INFOCOM2002*, vol:3, pages:1326-1335, June 2002.

[31] C. Huitema. IPv6: The new Internet Protocol. Prentice Hall, 1998.

[32] N. Gogate and S. S. Panwar. On resequencing model for high speed networks. *In 13th Proceedings. IEEE INFOCOM. Networking for global communication(INFOCOM'94)*. vol:1, pages:40-47, June 1994.

[33] Y. Wang and S. Lin. Error resilient video coding using multiple description motion compensation. *IEEE Transaction. Circuits and Systems for Video Technology*. vol:12, no.6, pages:438-452, June 2002.

[34] S. Mao, S. Lin, S. S. Panwar, and Y. Wang. Reliable transmission of video over ad hoc networks using automatic repeat request and multipath transport. *In 54th IEEE Vehicular Technology Conference, VTC Fall 2001*, vol:2, pages:615-619, October 2001.

[35] C. E. Perkins, E. M. Royer, and S. R. Das. Ad hoc on-demand distance vector (AODV) routing. IETF Internet draft, draft-ietf-manet-aodv-12.txt, November 2002.

[36] T. Clausen and P. Jacquet. Optimized Link State Routing Protocol (OLSR). Network Working Group, Request for Comments 3626, Project Hipercom, INRIA(October 2003), *http ://www.ietf.org/rfc/rfc3626.txt*. October 2003.

[37] D.B. Johnson, D.A. Maltz, J. Broch, DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks, Computer Science Department, Carnegie MellonUniversity(April 2003), *http://www.monarch.cs.cmu.edu/internet-drafts/draft-ietf-manet-dsr-09.txt*, April 2003.

[38] L.Anis. Unicast et Multicast dans les réseaux Ad hoc sans fil. Thèse de doctoat, université de Vesrsailles Saint-Quentin-En-Yevlines, Specialité : Informatique, 19 juillet 2002.

[39] R.E.Bellman. Dynamic programming. Princeton University Press, Princeton, 1957.

[40] L.R. Ford Jr., D.R. Fulkerson. Flows in Networks. Princeton University Press, 1962.

[41] C. E. Perkins and E. M. Royer. Ad Hoc On Demand. Distance Vector (AODV) Routing. draft-ietf-manet-. aodv-02.txt, Novembre1998.

[42] J.Broch, David B.Johnson,David A.Maltz. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. draft-ietf-manet-dsr-07.txt,Draftietf, February 2002.

[43] D.B. Johnson, D.A. Maltz. Dynamic Source Routing in Ad Hoc Wireless Networks. Mobile Computing, edited by Tomasz Imielinski and Hank Korth, Chapter 5, pages 153-181, Kluwer Academic Publishers, 1996.

[44] M.Gerla, X.Hong, G.Pei. Fisheye State Routing Protocol (FSR) for Ad Hoc Networks. Draft IETF, *.http://tools.ietf.org/id/draft-ietf-manet-fsr-02.txt,* December 2001.

[45] G.Pei, M.Gerla,T.W.Chen. Fisheye State Routing : A Routing Scheme for Ad Hoc Wireless Networks. In Preceedings of IEEE international Conference on Communications (ICC), vol :1, pages 70-74, new Orleans, LA,USA, June 2000.

[46] Z.J. Haas, M.R. Pearlman, and P.Samar. The interzone routing protocol (ierp) for ad hoc networks. IETF Internet-Draft,*http://tools.ietf.org/wg/manet/draft-ietf-manet-zone-ierp/draft-ietf-manet-zone-ierp-02-from-01.wdiff.html* July 2002.

[47] ZJ.Haas, M.R.Pearlman, and P.Samar. The Intrazone Routing Protocol (IARP) for Ad Hoc Networks.DRAFT IETF, draft-ietf-manet-zone-iarp-01.txt„ June 2001

[48] projet terminodes, http://www.terminodes.org/.

[49] V.Pack, S.Corson. Temporally-ordered Routing Algorithm (TORA). Draft IETF, *http://www.ietf.org/internet-drafts/draft-ietf-manet-tora-spec-04.txt*, July 2001.

[50] Z. J. Haas, et al., eds., IEEE Journal on Selected Areas in Communications, Special Issue on. Wireless Ad Hoc Networks, vol: 17, No 8, pages:1395-1414, August 1999.

[51] C.E. Perkins, E.M. Royer, S.R. Das. Quality of service for ad hoc on-demand distance vector routing. (work in progress), IETF Internet Draft, draft-ietf-manet-aodvQoS-00.txt, July 2000.

[52] M.K.Marina, S.R.Das. On-demand Multipath Distance Vector Routing in Ad Hoc Networks. *Proceedings of the International Conference for Network Procotols*, pages:14-23, November 2001.

[53] Z. Ye, S.V. Krishnamurthy and S.K. Tripathi, A framework for reliable routing in mobile ad hoc networks, *In Proceedings of IEEE INFOCOM, Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies* vol :1,pages:270-280, March–April 2003.

[54] S.-J. Lee and M. Gerla. Split Multipath Routing with Maximally Disjoint Paths in Ad Hoc Networks. *In Proceedings IEEE ICC '01*, vol: 10, pages 3201–3205, 2001.

[55] S.Mao. *Realtime multimedia transport using multiples paths.* These de doctor philosiphie en genie electrique, universite de Michigan, janvier 2004.

[56] S. Mao, D. Bushmitch, S.Narayanan, and S. S. Panwar. MRTP: A Multiflow Real-Time Transport Protocol for Ad Hoc Networks. *IEEE transactions on Multimedia*, vol: 8, NO. 2, page 356-369, Avril 2006.

[57] A. Munaretto, H. Badis, K. Al Agha, and G. Pujolle. A Link-state QoS Routing Protocol for Ad Hoc Networks. *In IEEE MWCN'02: International Workshop On Mobile and Wireless Communications Networks*, Stockholm, Sweden, September 2002.

[58] A. Veres, A. Campbell, M. Barry and L-H SUN. Supporting Service Differentation in Wireless Packet Networks Using Distributed Control. *IEEE Journal on Selected Areas in Communications*, vol 19, no10, October 2000.

[59] Amina Meraihi Naimi. *Délai et Routage dans les réseaux ad hoc 802.11.* Thèse,PhD Université de Versaille Saint-Quentin-En-Yvelines, 2005.

[60] H. Badis, A. Munaretto, K. Al Agha, and G. Pujolle. QoS for Ad hoc Networking Based on Multiple-Metric: Bandwidth and Delay. I*n IFIP MWCN'03: International Workshop On Mobile and Wireless Communications Networks*, Singapore, October 2003.

[61] M. Barry, A. T. Campbell and A. Veres. Distributed Control Algorithms for Service Differentiation in Wireless Packet Networks. *In IEEE INFOCOM 2001*, pages 582–590, 2001.

[62] M.Ozdemir and A. B.M.Donald. An M/MGI/1/K Queing Model for IEEE 802.11 Ad hoc Networks. *In PE-WASUN'04*, Italy, October 2004.

[63] O.Tickoo, and B.Sikdar. Queueing Analysis and Delay Mitigation in IEEE 802.11 Random Access MAC based Wireless Networks. *In IEEE Infocom,* 2004.

.
.
.
.
.