

N° d'ordre : 11/2010-M/inf

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**

**Ministère de l'enseignement Supérieur et de la Recherche Scientifique**

**Université des Sciences et de la Technologie Houari Boumediene**

**Faculté d'Electronique et d'Informatique**



**MEMOIRE**

Présenté pour l'obtention du diplôme de MAGISTER En Informatique

Spécialité : Informatique Mobile

Par : M<sup>lle</sup> AOUFI Souhila

**Sujet**

**QoS dans les réseaux  
maillés sans fil**

Soutenu publiquement le, 16 mai 2010 devant le jury composé de :

Mr- M. AHMED NACER, Professeur à l'USTHB, Président

Mr- D. TANDJAOUI, Maître de Recherche au CERIST, Directeur de mémoire

Mr- M. BENCHAIABA, Maître de conférence/ A l'USTHB, Examineur

Mr- O. NOUALI, Maître de Recherche au CERIST, Examineur

## الملخص

يمكن اعتبار الشبكات اللاسلكية المتشابكة معيار اقتصادي واعد في مجال الخدمات اللاسلكية عريضة النطاق. تتكون هذه الشبكة من عنصرين أساسيين: عُقد عملاء و عُقد موجهات. العُقد الموجهات تشكل بنية تحتية لاسلكية وتتفاعل مع الشبكات السلكية لتوفير خدمة الاتصال بالإنترنت إلى شبكة العملاء. إن الشبكات اللاسلكية المتشابكة تتلقى العديد من التدفقات في آن واحد والتي لها احتياجات مختلفة، ولكن نظرا لمحدودية قدرة القناة اللاسلكية، تأثير التداخل والعدد الكبير للمستخدمين فان تحسين إنتاجية الشبكة ودعم جودة الخدمات يمثلان الآن متطلبين هامين في مثل هذه الشبكات.

لقد ثبت أن استخدام عدة قنوات بدلا من قناة واحدة في الشبكات اللاسلكية متعددة القفزات كفيل بتحسين مرد ودية الشبكة بشكل كبير، ومع ذلك تجدر الإشارة إلى أن استعمال عدة واجهات في كل عقدة يتطلب التخطيط لتوزيع القنوات، كما يجب إرضاء جميع التدفقات المتنافسة حيث ينبغي أن تقسم موارد الشبكة بكفاءة و عدالة بين جميع التدفقات. وأخيرا ينبغي أن تأخذ خوارزمية توزيع القنوات بعين الاعتبار العبء المروري في كل وصلة، هذا العبء يتأثر ببروتوكولات التوجيه.

استنادا إلى آلية توزيع القنوات و التوجيه فان المشكل المثير للاهتمام هو: إيجاد خوارزمية توجيه حيث تعمل بشكل صحيح و فعال لفئة جودة الخدمة و من ناحية أخرى تقلل من التأثيرات السلبية على الفئات الأخرى في حركة المرور.

في عملنا هذا قمنا بدراسة مشكل توزيع القنوات في الشبكة اللاسلكية متعددة الواجهات متعددة القنوات مع تمايز الخدمات، اقترحنا تحسين بروتوكول " لأجل دعم جودة الخدمة. مساهماتنا هي كما يلي: أولا قمنا بتصميم جديد للتوجيه قادر على موازنة حركة المرور في الشبكة. ثانيا اقترحنا بروتوكول توجيه باستعمال عدة أشجار، شجرة لكل فئة حركة مرور، يمكن لهذا البروتوكول أن يتكيف تلقائيا مع التغيرات في عبء حركة المرور. ثالثا اقترحنا وضع بروتوكول توزيع القنوات مع تمايز الخدمات و الذي يمكن له أن يتكيف بشكل حيوي مع التغيرات في الأحمال المرورية.

إن تقييم أعمالنا هاته مقارنة مع البروتوكول الأساسي " اظهر لنا أن نهجنا أكثر كفاءة من حيث الوقت اللازم للاستجابة و الإنصاف بين التدفقات من كلتا الفئتين.

**المصطلحات:** توزيع القنوات، تفريق الخدمات، التوجيه، وجودة الخدمة.

## Résumé

Les réseaux maillés sans fil (WMNs) sont envisagés comme un paradigme économique viable et une technologie prometteuse pour l'accès ubiquitaire sans fil à haut débit. Les WMNs consistent en des nœuds clients et des nœuds routeurs, où les routeurs forment une infrastructure sans fil 'backbone' et interagissent avec les réseaux câblés pour offrir une connexion Internet sans fil multi-sauts aux nœuds clients. Les WMNs véhiculent de nombreux flux simultanés qui ont des exigences diverses. Cependant, en raison de la capacité du canal limitée et l'influence de l'interférence, l'amélioration du débit du réseau et le soutien des applications QoS sont devenus deux exigences critiques dans ce type de réseaux.

L'utilisation de canaux multiples au lieu d'un seul canal dans les réseaux sans fil multi-sauts peut améliorer significativement le débit du réseau. Cependant, il faut noter que (1) plusieurs interfaces radios dans chaque nœud exigent une planification d'attribution de canal. (2) Les flux concurrents doivent être satisfaits. Les ressources du réseau devraient être partagés de façon efficace et équitable entre tous les flux concurrents. (3) Enfin, un algorithme d'attribution de canal devrait prendre en considération la charge de trafic dans chaque lien qui est affecté par le routage. Sur la base de mécanisme d'attribution du canal et de routage, un problème intéressant est de trouver un algorithme de routage optimal tels que (1), il fonctionne correctement et efficacement pour la classe QoS, et en attendant (2) qu'il réduit les influences négatives à d'autres catégories de trafic.

Dans ce travail, nous étudions le problème d'attribution de canal dans un réseau maillé sans fil multi-radio multi-canaux avec différenciation de service. Nous proposons une amélioration du protocole 'Hyacinth' pour le soutien de la QoS. Nos contributions sont les suivantes : d'abord, nous concevons une nouvelle métrique de routages capable d'équilibrer la charge de trafic dans le réseau. En second lieu, nous proposons un protocole de routage à base de plusieurs arbres de routage, un arbre pour chaque classe de trafic, ce protocole peut s'adapter automatiquement aux changements de la charge de trafic dans chaque classe ainsi qu'aux défaillances du réseau. Troisièmement, nous présentons un nouveau protocole pour l'attribution de canal avec différenciation de service qui peut s'adapter dynamiquement avec la variation des charges de trafic. L'évaluation de performance de nos protocoles et la comparaison avec le protocole de base 'hyacinth', nous montrons que notre approche est beaucoup plus performante en terme de délai de réponse et équité entre les flux des deux classes.

**Mots clés :** Attribution du canal, Différenciation de service, routage, QoS.

## Abstract

Wireless Mesh Networking (WMN) is envisioned as an economically viable paradigm and a promising technology in providing wireless broadband services. WMNs consist of mesh clients and mesh routers, where the mesh routers form a wireless infrastructure/backbone and interwork with the wired networks to provide multihop wireless Internet connectivity to the mesh clients. The WMNs carry many flows simultaneous that have different requirements. However, Due to the limited channel capacity, the influence of interference, the large number of users and the emergence of real-time multimedia applications, improving network throughput and supporting QoS have become two critical requirements in such networks

Using multiple channels instead of a single channel in multihop wireless networks has been shown to be able to improve the network throughput dramatically. However, it is noted that (1), several radio interfaces in each node requires a planning of channel assignment. (2) Flows competitors must be satisfied. Network resources should be shared efficiently and fairly among all competing flows. (3) Finally, an algorithm for channel assignment should take into consideration the traffic load in each link is affected by routing. Based on the mechanism of channel allocation and routing, an interesting problem is to find an optimal routing algorithm such that (1), it works correctly and efficiently for QoS class, and meanwhile (2) it reduces the negative influences to other traffic categories.

In this work, we study the problem of channel assignment in a wireless mesh network multi-radio multi-channel with service differentiation. We propose an improved protocol 'Hyacinth' for support of QoS. Our contributions are as follows: First, we design a new routing metric capable of balancing the traffic load in the network. Secondly, we propose a routing protocol based on multiple routing trees, one tree for each traffic class, this protocol can automatically adapt to changes in traffic load in each class as well as network failures. Third, we present a new protocol for channel allocation with service differentiation that can adapt dynamically to changes in traffic loads. The performance evaluation of our protocols and comparison with the basic protocol 'hyacinth', we show that our approach is much more efficient in terms of response time and fairness among flows of the two classes.

**Keywords :** channel assignment, service differentiation, routing, QoS.

## *Dédicace*

*Je dédie ce modeste travail*

- ❖ *A mes chers parents pour leur éternel amour, leurs sacrifices et leur patience*
  
- ❖ *A mon sincère mari Ali pour son soutien et ses encouragements*
  
- ❖ *A mes chers frères et sœurs qui m'ont toujours assisté*
  
- ❖ *A mes chers enfants, serine et mohamed, en lui souhaitant le succès dans leur vie.*
  
- ❖ *A tous mes amis et mes collègues*
  
- ❖ *A tous les gents pour qui le savoir est précieux*

## ***Remerciements***

Mes vifs remerciements à DIEU pour tous ses dons et les bénédictions de toute ma vie.

Je remercie énormément le professeur Nadjib BADACHE, directeur du Centre de Recherche sur l'Information Scientifique et Technique (CERIST) qui m'a donnée l'opportunité de travailler dans le domaine des réseaux sans fil.

Je présente mes remerciements à Dr. Djamel Tandjaoui, qui a accepté de m'encadrer dans le cadre de mon projet de magistère et pour le sujet particulièrement intéressant.

Je remercie vivement le professeur Mohamed AHMED NACER de l'USTHB d'avoir accepté de présider le juré de ma soutenance.

Je tiens aussi à remercier les membres du jury : Dr. Mahfoud BENCHAIBA et Dr. Omar NOUALI pour le temps et l'effort investis pour juger mon travail.

Je voudrais remercier vivement tous les membres du CERIST pour leur aide. Je remercie spécialement Dr. Omar NOUALI qui m'a accueilli au sein de sa division et qui m'a permis de travailler sur des problèmes très intéressants.

J'aimerais également remercier Mr. Ashish RANIWALA pour ses explications précieuses qui m'ont permis d'enrichir mon travail et de le perfectionner.

Je suis reconnaissante à mon frère Abdelkader pour son encouragement, aide et pour sa lecture du mémoire et ses remarques constructives.

Je remercie énormément M<sup>lle</sup> Djedjiga MOUHEB pour sa disponibilité et ses encouragements tout au long de mon projet.

# Table des matières

<b>INTRODUCTION GENERALE .....</b>	<b>1</b>
<b>PARTIE 1: ETAT DE L'ART</b>	
<b>CHAPITRE 1: LES RESEAUX MAILLES SANS FIL .....</b>	<b>5</b>
1.1 INTRODUCTION.....	5
1.2 DEFINITION .....	5
1.3 TOPOLOGIES.....	6
1.4 CARACTERISTIQUES .....	8
1.5 AVANTAGES.....	8
1.6 CONTRAINTES DE BASE .....	9
1.7 APPLICATIONS.....	9
1.8 CONCLUSION.....	12
<b>CHAPITRE 2: LE ROUTAGE DANS LES RESEAUX MAILLES SANS FIL .....</b>	<b>13</b>
2.1 INTRODUCTION.....	13
2.2 METRIQUES DE ROUTAGE .....	13
2.2.1 Métriques de routage avec un seul canal.....	14
2.2.2 Métriques avec plusieurs canaux .....	16
2.3 ROUTAGE AU MIEUX .....	18
2.3.1 Protocoles réactifs.....	18
2.3.2 Protocoles pro-actifs.....	19
2.4 ROUTAGE AVEC QUALITE DE SERVICE (QoS).....	20
2.5.1 Protocoles de routage Général .....	21
2.5.2 Routage à base de localisation.....	23
2.5.3 Routage avec Différenciation de Service .....	23
2.5.4 Routage avec contraintes multiples.....	25
2.5.5 Routage avec canaux multiples .....	25
2.5.6 Routage avec multi-chemin .....	26
2.5.7 Routage avec puissance de transmission .....	26
2.5.8 Synthèse.....	27
1.5 CONCLUSION .....	28

<b>CHAPITRE 3: ATTRIBUTION DE CANAL .....</b>	<b>29</b>
3.1 INTRODUCTION .....	29
3.2 GESTION DES RESSOURCES DANS UN WMN MULTI-RADIO MULTI-CANAU .....	30
3.3 ATTRIBUTION DU CANAL DANS RESEAUX CELLULAIRES / RESEAUX MAILLES SANS FIL..	30
3.4 PRELIMINAIRES.....	32
3.4.1 <i>Le graphe de Connectivité</i> .....	32
3.4.2 <i>Le graphe de Conflit</i> .....	33
3.4.3 <i>Le graphe de Conflit Multi-radio</i> .....	35
3.5 LES CONTRAINTES ET LES DEFIS DANS L'ATTRIBUTION DE CANAL .....	35
3.6 ATTRIBUTION DE CANAL ET TOPOLOGIE DU RESEAU .....	38
3.7 ATTRIBUTION DU CANAL, CONNECTIVITE ET INTERFERENCE.....	39
3.8 CLASSIFICATION D'ATTRIBUTION DE CANAL .....	41
3.8.1 <i>Statique</i> .....	41
3.8.2 <i>Dynamique</i> .....	42
3.8.3 <i>Hybride</i> .....	42
3.8.4 <i>Résumé des stratégies d'attribution de canal</i> .....	43
3.9 TRAVAUX LIES .....	44
3.9.1 <i>MR-LQSR</i> .....	44
3.9.2 <i>Hyacinth</i> .....	44
3.9.3 <i>MCR</i> .....	45
3.9.4 <i>AODV-ST</i> .....	46
3.9.5 <i>MesTic</i> .....	47
3.9.6 <i>Attribution de canal avec QoS</i> .....	48
3.9.7 <i>Synthèse des protocoles MC-MR</i> .....	48
3.5 CONCLUSION .....	49

## **PARTIE 2: CONTRIBUTION**

<b>CHAPITRE 4: ATTRIBUTION DE CANAL AVEC DIFFERENTIATION DE SERVICE .....</b>	<b>51</b>
4.1 INTRODUCTION .....	51
4.2 MOTIVATION .....	52
4.3 ARCHITECTURE DU SYSTEME ET FORMULATION DE PROBLEME.....	54
4.3.1 <i>Architecture du système</i> .....	54

4.3.2	<i>Objectif de l'attribution de canal</i> .....	54
4.3.3	<i>Formulation de problème</i> .....	55
4.4	ALGORITHME D'ATTRIBUTION DE CANAL .....	60
4.4.1	<i>Avantage de topologie en arbre</i> .....	60
4.4.2	<i>Construction des arbres de routage et attribution de canal</i> .....	61
4.4.3	<i>Equilibrage de charge</i> .....	76
4.4.4	<i>Tolérance aux pannes</i> .....	80
4.5	CONCLUSION .....	82
<b>CHAPITRE 5: SIMULATION ET RESULTATS .....</b>		<b>83</b>
5.1.	INTRODUCTION .....	83
5.2.	PRESENTATION DU SIMULATEUR NS2 .....	83
5.3.	SUPPORT DU MULTI-INTERFACE, MULTI-CANAU DANS NS2 .....	84
5.4.	IMPLEMENTATION DU PROTOCOLE CA-DIFFSERV PROPOSE .....	86
5.4.1.	<i>Modélisation du protocole sous forme évènementielle</i> .....	87
5.4.2.	<i>Réalisation des testes et interprétation des résultats</i> .....	88
5.5.	EVALUATION DE PERFORMANCE .....	88
5.5.1.	<i>Métrique de construction de l'arbre de routage</i> .....	88
5.5.2.	<i>Métrique de performance</i> .....	89
5.5.3.	<i>Scénario de simulation</i> .....	90
5.5.4.	<i>Analyse des résultats</i> .....	91
5.6.	CONCLUSION .....	94
<b>CONCLUSION GENERALE .....</b>		<b>95</b>
<b>REFERENCES BIBLIOGRAPHIQUES.....</b>		<b>108</b>
<b>ANNEXES</b>		
<b>ANNEXE 1 : LE SIMULATEUR NS .....</b>		<b>104</b>
<b>ANNEXE 2 : SUPPORT DU MULTI INTERFACE DANS NS2.33 .....</b>		<b>106</b>
<b>ANNEXE 3 : DESCRIPTION DES MESSAGES ECHANGES.....</b>		<b>111</b>
<b>ANNEXE 4 : FICHER DE TRACE ET LE LANGAGE AWK.....</b>		<b>120</b>

## Liste des figures

Figure 1.1 : Réseau Sans-fil en Backbone.....	6
Figure 1.2 : Réseau Sans-fil de Clients .....	7
Figure 1.3 : Réseau Sans-fil Hybride .....	7
Figure 1.4 : Réseau Domestique .....	10
Figure 1.5 : Réseau de Communauté .....	10
Figure 1.6 : Réseau de Véhicule.....	11
Figure 1.7 : Réseau d'entreprise.....	11
Figure 2.1 : Classification des protocoles .....	18
Figure 2.2 : Principe de fonctionnement de MPR-OD.....	22
Figure 3.1 : Réutilisation de canal dans les réseaux cellulaires .....	31
Figure 3.2 : Exemple de graphe de connectivité .....	33
Figure 3.3 : Exemple de graphe de conflit .....	34
Figure 3.4 : Exemple de graphe de conflit multi-radio .....	35
Figure 3.5 : Dépendance de canal .....	36
Figure 3.6 : Stratégies pour attribution de canal sensible à la charge de trafic.....	37
Figure 3.7 : Topologie du réseau avec variation d'attribution de canal.....	38
Figure 3.8 : Compromis entre la connectivité et l'interférence.....	40
Figure 3.9 : Classification des stratégies d'attribution de canal.....	43
Figure 4.1 : Architecture du réseau à trois niveaux.....	51
Figure 4.2 : Contrainte d'interférence .....	57
Figure 4.3 : Topologie de WMN en arbre .....	57
Figure 4.4 : Problème d'optimisation.....	60
Figure 4.5 : Entrés/Sorties du processus de construction des arbres de routage.....	63
Figure 4.6 : Processus de construction des arbres de routage .....	65
Figure 4.7 : Exemple d'arbres de routage .....	66
Figure 4.8 : Entrés/Sorties du processus d'attribution de canal .....	70
Figure 4.9 : Les E/S du processus CA pour les interfaces radio de la passerelle.....	71
Figure 4.10 : Processus d'attribution des canaux pour la passerelle .....	72
Figure 4.11 : Les e/s du processus CA pour les interfaces radio d'un nœud intermédiaire .....	73
Figure 4.12 : Processus d'attribution des canaux pour un nœud intermédiaire .....	74
Figure 4.13 : Exemple d'attribution de canal .....	75

Figure 4.14 : Processus de réattribution de canal.....	77
Figure 4.15 : Processus de reconstruction de l'arbre de routage.....	79
Figure 4.17 : Processus de détection et de rétablissement de pannes .....	81
Figure 5.1 : Architecture d'un nœud mobile .....	85
Figure 5.2 : Architecture modifier d'un nœud mobile, avec support de multi interface.....	86
Figure 5.3 : Topologie grille d'un réseau constitué de 9 nœuds .....	88
Figure 5.4 : Arbre de routage construit par les deux algorithmes .....	89
Figure 5.5 : Latence moyenne (CA-DiffServ Vs. Hyacinth) .....	91
Figure 5.6 : Latence moyenne (CA-DiffServ Vs. DSDV) .....	92
Figure 5.7 : Equité en latence (CA-DiffServ Vs. Hyacinth) .....	92
Figure 5.8 : Latence en fonction de nombre de canaux .....	93
Figure 5.9 : Overhead en terme de nombre de paquets de contrôle .....	94
Figure 5.10 : Overhead en terme de charge de routage normalisé.....	94

## Liste des tableaux

Tableau 2.1 : Caractéristiques principales des métriques de routage.....	17
Tableau 2.2 : Approche de routage avec QoS dans les réseaux mesh .....	27
Tableau 3.1 : Comparaison des stratégies d’attribution du canal.....	43
Tableau 3.2 : Etude comparative des protocoles d’attribution de canal existants.....	49
Tableau 4.1 : Table des notations.....	55
Tableau 4.2 : Table des contraintes et objectifs de notre problème .....	59
Tableau 4.3 : Table des contraintes et objectifs de notre problème .....	60
Tableau 4.4 : Table des messages utilisés dans la construction de l’arbre de routage.....	64
Tableau 4.5 : Table des messages échangés dans l’attribution de canal .....	71
Tableau 4.6 : Liste des canaux triés dans chaque ensemble pour chaque nœud .....	75
Tableau 4.7 : Table des messages échangés dans la reconstruction des arbres .....	80

---

# Introduction générale

---

Les progrès technologiques récents ont permis l'apparition d'une grande variété de nouveaux moyens permettant à un utilisateur d'accéder et d'utiliser l'information qui l'intéresse en tout lieu et à tout moment.

Actuellement, le déploiement des réseaux maillés sans fil (Wireless Mesh Network WMN) est considéré comme un pas prometteur vers l'accès ubiquitaire sans fil à haut débit. Ces réseaux fournissent une large connectivité et requièrent un coût très réduit par rapport au réseau câblé. En plus de cet avantage, les réseaux maillés sans fil sont faciles à maintenir, ils sont robustes et fournissent des services fiables. Ils peuvent être déployés dans plusieurs endroits et couvrir des zones difficiles d'accès aux câbles.

Par ailleurs, la popularité et le succès du Wi-fi (*Wireless Fidelity*) ont suscité l'intérêt à développer et étendre des réseaux sans-fil sur des zones larges. De tels points d'accès (AP) sont disponibles mais le coût du déploiement de ces réseaux qui les utilisent est rehaussé par le coût d'interconnexion des points d'accès et de leurs connexions à Internet définissant ainsi le réseau *Backhaul*. Le *Backhaul* sans-fil est apparu comme une alternative intéressante qui permet de réduire énormément les coûts de déploiement. Des connexions radio seront établies entre la plupart des points d'accès et seulement quelques connexions filaires seront déployées à Internet pour servir tout le réseau.

Chaque nœud du réseau possède deux fonctions essentielles. En effet, un nœud doit servir ses clients locaux et relayer en plus de son trafic, celui qui est reçu des autres nœuds pour que tous les nœuds soient servis dans les deux sens, venant de/allant à la passerelle Internet. La charge du trafic sur le *Backhaul* dépend énormément de la dimension du réseau, la topologie utilisée, la bande passante, les techniques de modulation et de routage.

Les exigences des applications multimédia laissent penser que certaines applications pourraient tirer parti de certaines garanties qui pourraient être offertes par ce type de réseau. Par exemple, garantir une borne sur le délai de transmission des paquets peut être profitable aux applications de téléphonie ; garantir un débit peut être nécessaire pour les applications de

vidéo à la demande, etc. Les WMNs véhiculent de nombreux flux simultanés qui ont des exigences diverses. Cependant, en raison de la capacité du canal limitée et l'influence de l'interférence, l'amélioration du débit du réseau et le soutien des application QoS sont devenus deux exigences critiques dans ce type de réseaux.

L'équipement de chaque nœud par plusieurs interfaces réseaux améliore considérablement le débit du réseau [25] [26]. Les normes IEEE 802.11b/g et IEEE 802.11a offrent 3 et 12 canaux non chevauchés, respectivement, ce qui peut être utilisé simultanément par un routeur mesh pour la transmission et la réception dans un voisinage en assignant des canaux non-chevauchés aux différentes interfaces radios. Ceci nous amène à une utilisation efficace du spectre et une augmentation de la bande passante effective du réseau. Cependant, il faut noter que :

- Plusieurs interfaces radios dans chaque nœud exigent une planification d'attribution de canal. Les canaux doivent être affectés de telle manière, que l'interférence soit diminuée et la connectivité des nœuds soit maintenue en même temps.
- Les flux concurrents doivent être satisfaits. Les ressources du réseau devraient être partagées de façon efficace et équitable entre tous les flux concurrents.
- Enfin, un algorithme d'attribution de canal devrait prendre en considération la charge de trafic dans chaque lien qui est affecté par le routage. D'une manière générale, une large bande passante devra être accordée aux nœuds qui ont plus de trafic. Ces stratégies d'attribution de canal consciente du trafic distribueront les ressources radio de manière à correspondre à la répartition de la charge de trafic dans le réseau.

Dans notre travail, nous traitons le problème d'attribution du canal dynamique dans les réseaux maillés sans fil avec Différentiation de Service. Nous proposons une amélioration du protocole 'Hyacinth' pour le soutien de la QoS. Nous considérons deux classes de trafic : Best Effort (BE) pour le trafic traditionnel et QoS pour le trafic qui a des exigences de QoS. Nos contributions s'articulent dans les points suivants:

1. Une nouvelle métrique de routages capable d'équilibrer la charge de trafic dans le réseau. Cette métrique est fonction de quatre paramètres, trois sont statiques, à savoir le nombre d'interfaces radio que dispose un nœud, le nombre de voisins directs d'un nœud ainsi que la distance entre le nœud routeur et la passerelle Internet. Et un dynamique qui est la charge de trafic qui passe par un nœud.

2. Un algorithme de routage à base de plusieurs arbres recouvrants, un arbre pour chaque classe de trafic, avec équilibrage de charge. Cet algorithme peut s'adapter automatiquement aux changements de charge du trafic dans les différentes classes ainsi qu'aux défaillances du réseau.
3. Un algorithme d'attribution de canal avec différenciation de service distribué. Cet algorithme apporte aux nœuds du réseau une connaissance sur la charge des canaux utilisés dans leur voisinage ainsi que le type de trafic qui transporte (QoS ou BE). Cette connaissance est suffisante pour leur permettre de déterminer les canaux les moins utilisés pour les attribuer à leurs interfaces radios. Cet algorithme peut s'adapter dynamiquement aux charges de trafic dans chaque classe.

Pour évaluer les performances de notre protocole, d'attribution du canal et de routage, baptisé 'CA-DiffServ', nous avons effectué une série de tests et de comparaisons avec le protocole Hyacinth [26] comme exemple de protocole multi-canaux multi-interfaces et le protocole DSDV comme exemple de protocole mono-canal mono-interface pour générer les graphes de latence, d'équité et ceux d'overhead. Les résultats obtenus ont permis de valider le fonctionnement de notre protocole. Ils indiquent en particulier comment le protocole réagit face à la coexistence du trafic Best Effort avec le trafic QoS.

Ce document s'articule autour de cinq chapitres : dans le premier chapitre, nous introduisons les concepts des réseaux maillés sans fil, nous décrivons brièvement le WMN, ses caractéristiques, ses avantages, ses contraintes de base et ses domaines d'applications.

Nous présentons dans le second chapitre un état de l'art sur les métriques de routage et nous proposons une taxonomie de protocoles de routage consacrés aux réseaux maillés sans fil.

Le troisième chapitre est consacré pour détailler le problème d'attribution de canal dans les réseaux maillés sans fil. Nous y mettons en évidence les contraintes associées et les défis d'attribution de canal. Nous classifions par la suite les algorithmes d'attribution de canal selon plusieurs critères et nous présentons les travaux faits dans ce domaine.

Notre proposition d'un protocole d'attribution du canal avec différenciation de service est exposée dans le quatrième chapitre. Après avoir présenté nos motivations pour ce choix, nous décrivons l'ensemble des extensions nécessaires à effectuer sur le protocole 'Hyacinth' afin de tenir compte de la classe de trafic dans l'attribution du canal et le calcul des routes.

Le dernier chapitre montre l'outil de simulation Network Simulator (NS-2) et décrit l'ensemble des tests et des comparaisons réalisés.

Une conclusion générale et des perspectives font la fin de ce mémoire

PARTIE I

ETAT DE L'ART

---

# Les réseaux maillés sans fil

---

## 1.1 Introduction

Dans ce chapitre, nous allons aborder les notions de base pour un réseau maillé sans-fil, en commençant par la définition et les différentes topologies. Nous présenterons ensuite les caractéristiques, avantages et les domaines d'applications du réseau maillés sans fil.

## 1.2 Définition

Les réseaux maillés sans fil ou bien les réseaux mesh appartiennent à une classe de réseaux conçus à l'image des réseaux filaires, basés sur le principe des communications multi-sauts et dans lesquels la gestion est totalement distribuée. L'architecture d'un réseau maillé sans fil est composée de deux types de nœuds [1] : les nœuds routeurs et les nœuds clients.

Les nœuds routeurs forment le backbone. Ils sont fixes ou d'une mobilité très réduite. Ils n'ont pas de contraintes d'énergie, et permettent de maintenir la connectivité entre différents clients. Un nœud routeur équipé d'une passerelle et d'un pont, peut intégrer des fonctionnalités d'accès à plusieurs réseaux assurant ainsi l'interconnexion entre eux. Les nœuds clients forment le réseau client (tel que les PC, Notebooks, ...). Ils peuvent être mobiles ou stationnaires ; connectés de façon directe aux routeurs ou indirecte par le biais d'une connexion proche en proche à travers d'autres clients (en mode Ad hoc).

Les réseaux maillés sans fil ont plusieurs caractéristiques en commun avec les réseaux Ad hoc (multi-sauts, auto-organisation, ...) et d'autres différentes (notamment le backbone, le rapport à l'énergie, et la façon de considérer la mobilité). Ces différences sont importantes et doivent être prises en compte lors de l'élaboration d'un protocole de routage.

### 1.3 Topologies

Comme nous l'avons déjà noté, les nœuds se répartissent en deux types: des nœuds routeurs et des nœuds clients. Les nœuds clients peuvent, dans certaines topologies, jouer le rôle de relais pour les paquets des autres nœuds.

Il existe trois groupes essentiels de topologies *mesh* présentés ci-après [2]:

#### 1. Les réseaux maillés sans-fil en Backbone (ou Backhaul)

L'architecture est illustrée dans la figure 1.1, où les lignes continues et en pointillées désignent respectivement les liens filaires et sans-fil. Ce type de réseaux intègre des points d'accès formant une infrastructure à laquelle se connectent les clients. Différents types de technologies radio sont utilisées pour le *backbone*, particulièrement les technologies IEEE 802.11. La fonctionnalité de passerelle (*gateway*) intégrée dans certains points d'accès assure la connectivité à Internet. Cette approche de maillage crée un *backbone* pour les clients et permet de faire co-exister les réseaux *mesh* sans-fil avec les réseaux classiques. Les clients sont capables de communiquer directement avec les points d'accès s'ils utilisent la même technologie radio, sinon ils communiquent avec les stations de base liées filairement au *backbone* sans-fil.

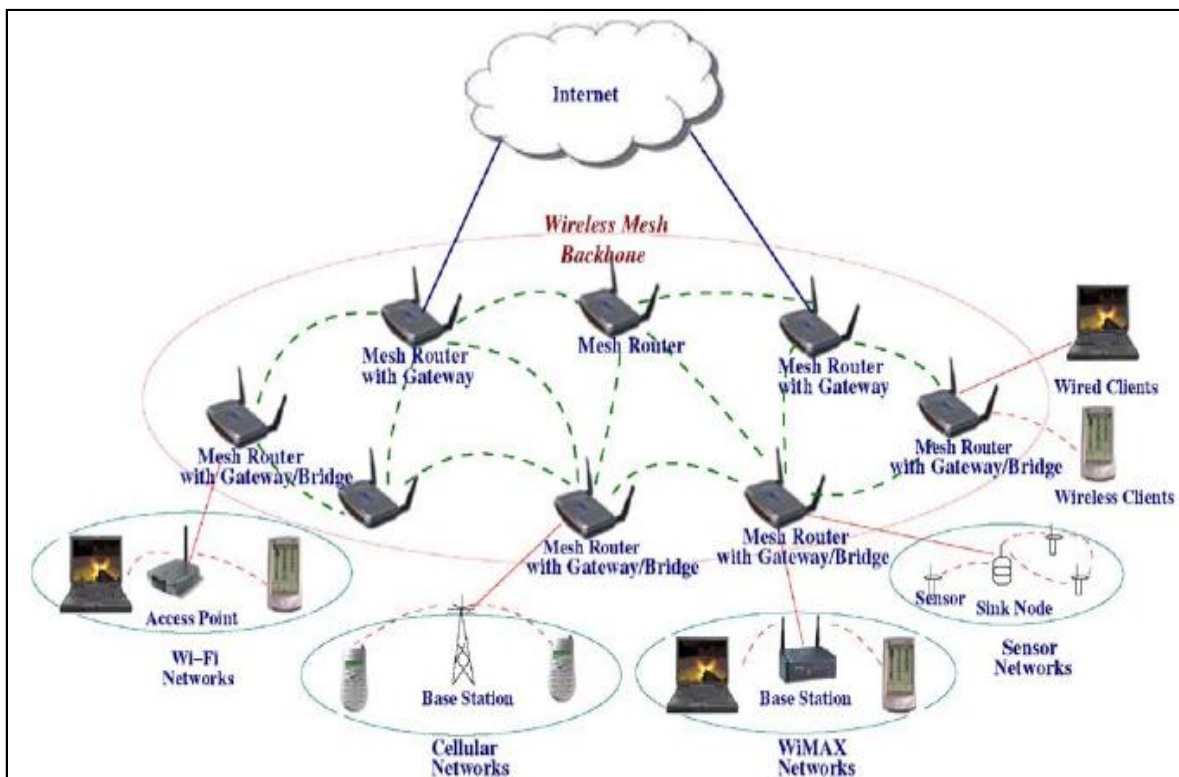


Figure 1.1 : Réseau Sans-fil en Backbone

### 2. Les réseaux maillés sans-fil de clients (ou Ad hoc)

Le maillage des clients crée un réseau pair à pair. Dans ce type d'architecture, les nœuds client constituent le réseau et assurent les fonctionnalités de routage, configuration et exécution des applications. Cette structure est détaillée dans la figure 1.2. Un paquet envoyé par un nœud doit effectuer des sauts d'un nœud à un autre pour arriver à sa destination. En général, ce réseau est formé en utilisant un seul type d'interface radio.

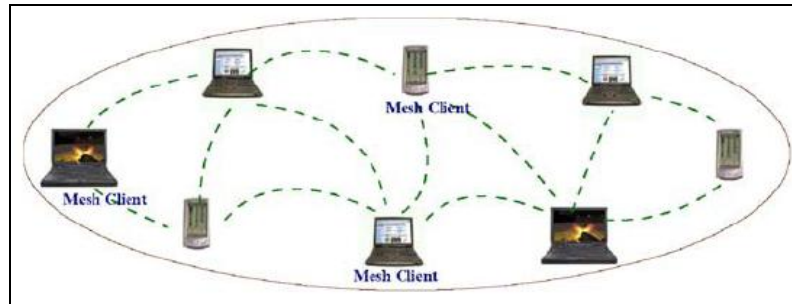


Figure 1.2 : Réseau Sans-fil de Clients

### 3. Les réseaux maillés sans-fil hybrides

Cette structure est la combinaison des deux types de réseaux décrits précédemment. Dans la figure 1.3, les clients peuvent accéder au réseau soit par l'intermédiaire des points d'accès ou bien par le maillage avec d'autres nœuds clients. Le *backbone* assure la connectivité à d'autres types de réseaux comme Internet, Wi-fi, WiMAX, les réseaux cellulaires ou de capteurs, tandis que la capacité de routage des nœuds clients développe la connectivité et la couverture à l'intérieur du réseau des clients.

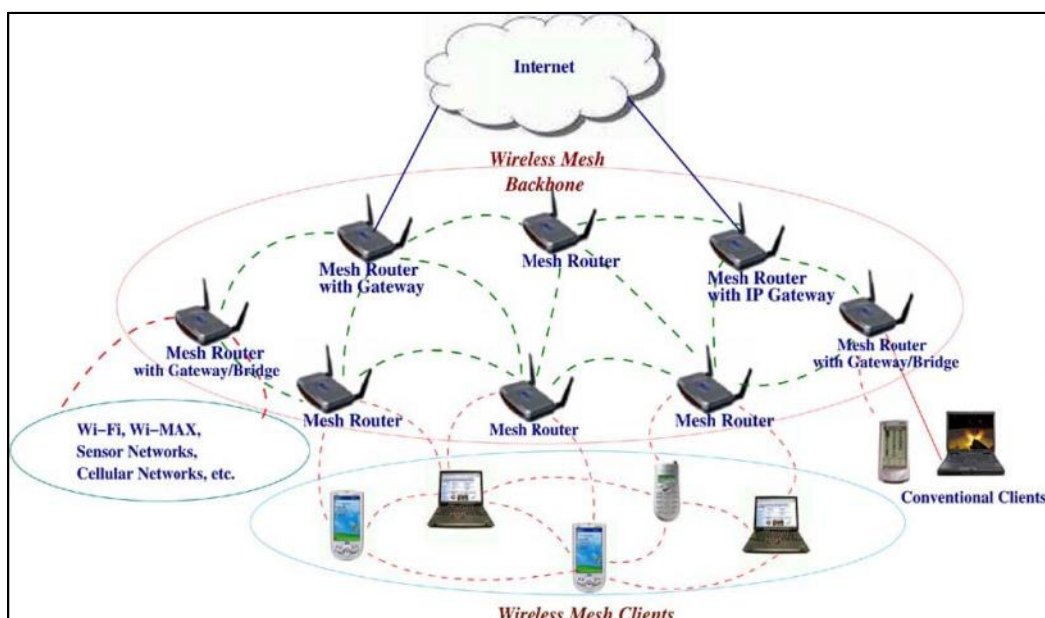


Figure 1.3 : Réseau Sans-fil Hybride

## 1.4 Caractéristiques

Les caractéristiques des réseaux maillés sans-fil sont détaillées ci-dessous.

### 1. *Un Réseau en Multi-sauts*

L'objectif est de développer les réseaux maillés sans-fil et d'étendre la couverture des réseaux courants sans sacrifier la capacité du canal. Un autre objectif est d'assurer une connectivité sans visibilité directe (*Non-Line-Of-Sight* NLOS) entre les nœuds. Pour pouvoir satisfaire ces propriétés, le maillage en multi-sauts est indispensable afin d'atteindre des bons débits sans sacrifier la couverture par des liens plus courts et réduire l'interférence entre les nœuds.

### 2. *La Mobilité*

Les points d'accès possèdent une mobilité très réduite (dans la plupart des cas, ils sont fixes). Les nœuds clients peuvent être stationnaires ou mobiles (PC portables, PDAs, PocketPCs, téléphones mobiles, etc.).

### 3. *Divers Types d'Accès au Réseau*

Dans les réseaux maillés sans-fil, les communications *backbone* à Internet et « pair to pair » sont supportées. En plus, l'intégration des réseaux maillés sans-fil à d'autres réseaux permet de fournir des nouveaux services à celles-ci.

### 4. *Compatibilité et Interopérabilité avec les réseaux sans-fil existants*

Les réseaux maillés sans fil basés sur les technologies IEEE 802.11 doivent être compatibles avec les normes IEEE 802.11. L'objectif visé est de desservir les clients conventionnels Wi-fi et adapter la technique de maillage. De tels réseaux doivent inter-opérer avec d'autres technologies sans-fil comme le WiMAX (*Worldwide Interoperability for Microwaves Access*) et les réseaux cellulaires.

## 1.5 Avantages

Voici les principaux avantages des réseaux mesh sans-fil :

- Déploiement simple et rapide : les réseaux maillés sont considérés comme faciles, simples et économiques à déployer par rapport aux réseaux Wi-Fi traditionnels. Ces derniers nécessitent l'irrigation en accès en réseau filaire de chacun des points d'accès sans fil. Cependant, dans un réseau maillé, certains points d'accès servent de passerelle pour d'autres points d'accès dépourvus d'accès filaire au réseau ;
- Coût d'installation et de maintenance réduit ; sur le moyen terme, l'investissement est facilement rentabilisé ;

- Capacité de passage à l'échelle : les réseaux sans fil peuvent être dimensionnés au plus juste et suivre simplement l'évolution des besoins ;
- Flexibilité et souplesse : le déploiement de ce réseau permet l'interconnexion de plusieurs d'autres types de réseaux à savoir : les réseaux cellulaires, filaires, wimax et les réseaux wifi ;
- Des possibilités d'alimentation diverses dues à la stationnarité des nœuds routeurs.

## 1.6 Contraintes de base

Les réseaux maillés sans fil ont deux caractéristiques principales :

- les nœuds routeurs sont statiques alors que les nœuds clients peuvent être mobiles.
- le moyen sans fil est de nature partagée. La propagation radio dans l'air est soumise à bon nombre de contraintes spécifiques. Nous avons déjà entrevu au paragraphe précédent les problèmes de fiabilité dus à l'environnement d'utilisation. Une variation *a priori* anodine dans l'environnement d'un nœud peut se traduire par la disparition d'un lien dans le réseau. D'autre part, le médium radio est partagé. Ce qui signifie que lorsqu'un nœud émet un signal, tous les nœuds dans un certain périmètre ne pourront pas transmettre de données simultanément, sous peine de provoquer des interférences à la réception.

## 1.7 Applications

La recherche et le développement dans le domaine des réseaux maillés sans-fil sont appuyés sur diverses applications. Ces applications ne peuvent pas être directement supportées par les réseaux sans-fil existants comme les réseaux cellulaires, capteurs, ou par le standard 802.11, etc. Nous discutons ci-après quelques unes de ces applications [2].

### a) Réseau Domestique (*Broadband Home Networking*)

Actuellement, le réseau domestique est réalisé par les technologies WLAN (*Wireless Local Area Network*) IEEE 802.11. Un problème qui surgit toujours est les localisations des points d'accès. Sans une étape préalable de planification, une maison (même très petite) a plusieurs zones mortes sans couverture radio. Les solutions basées sur la planification sont coûteuses ainsi que la méthode de déploiement de plusieurs points d'accès risque de ne pas couvrir tout l'espace. La solution du réseau maillé sans-fil est capable de résoudre ce problème à moindre coût (voir figure 1.4). En effet, remplacer les points d'accès classiques (avec interface filaire) par des points d'accès *mesh* permet de réduire les installations filaires. Ces points d'accès constituent par conséquent le *backbone* auquel se connectent tous les équipements sans-fil de la maison. Les zones mortes peuvent être éliminées par l'ajout de points d'accès, changer

leurs localisations (plus facile avec les points d'accès *mesh*) ou ajuster leurs niveaux de puissance.

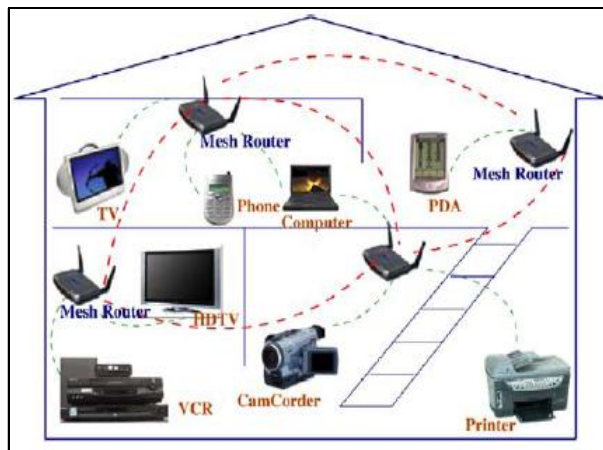


Figure 1.4 : Réseau Domestique

b) Réseau de Communauté et de Voisinage (*Community and Neighborhood Networking*)

Dans une communauté, l'architecture de réseau courante se base sur le DSL (*Digital Subscriber Line*) ou sur le câble de connexion au réseau Internet, et seulement le dernier saut reste sans-fil. Ce type de réseau possède plusieurs inconvénients qui sont essentiellement :

- Le passage obligatoire par Internet pour se partager et s'échanger des fichiers ;
- L'espace large entre les maisons non-couvertes par les services sans-fil, et
- L'existence d'une seule passerelle pour se connecter à Internet pour chaque maison.

La flexibilité du réseau maillé (voir figure 1.5) permet d'éviter ces problèmes. En effet, le *backbone* sans-fil entre les maisons garantit le partage d'applications et de données sans avoir besoin de passer par Internet. En plus, il donne la possibilité à une maison de se connecter à Internet moyennant une des passerelles disponibles sur le réseau de voisinage.

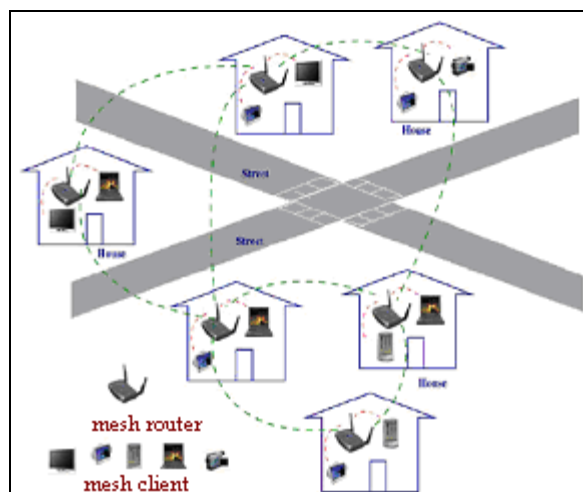


Figure 1.5 : Réseau de Communauté

c) Réseau de Véhicule (*Transportation System*)

Au lieu de limiter l'accès IEEE 802.11 et 802.16 à des stations fixes, on est capable d'étendre l'accès en utilisant le maillage aux trains, les bus et les bateaux. Pour un tel accès, deux points essentiels doivent être résolus.

- Un accès haut-débit au *backbone* sans-fil à partir du véhicule vers Internet, et
- L'existence de réseaux maillés dans le véhicule, comme il est montré dans la figure 1.6.

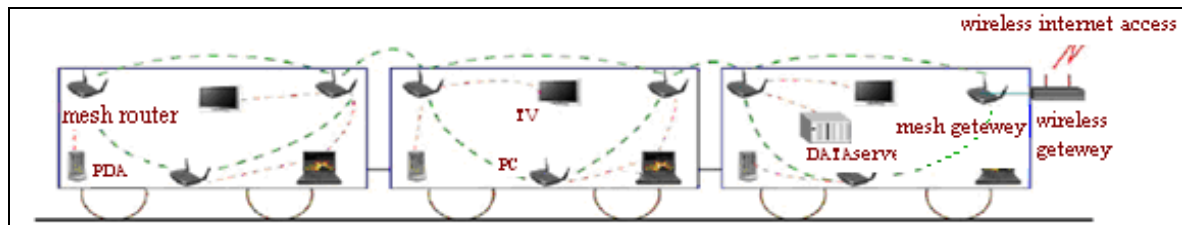


Figure 1.6 : Réseau de Véhicule

## d) Réseau d'entreprise

Cela peut être un petit réseau dans un bureau ou un réseau de taille moyenne pour tous les bureaux dans un immeuble, ou un réseau de grande échelle entre les bureaux dans plusieurs bâtiments. Actuellement, la norme IEEE 802.11 des réseaux sans fil est largement utilisée dans différents bureaux. Toutefois, ces réseaux sans fil sont des îles toujours isolées. Les connexions entre eux sont faites par des connexions Ethernet câblées, qui sont la principale raison du coût élevé des réseaux d'entreprises. Si les points d'accès sont remplacés par des routeurs mesh, comme il est montré dans la Figure 1.7, les câbles Ethernet peuvent être éliminés. Les réseaux maillés sans fil peuvent grandir facilement si la taille de l'entreprise augmente. Les réseaux maillés sans fil pour les réseaux d'entreprises sont beaucoup plus compliqués que pour la maison parce que plusieurs nœuds et plusieurs topologies compliqués de réseau sont impliqués. Le modèle de service de réseau d'entreprise peut être appliqué dans plusieurs domaines comme les aéroports, centres commerciaux, centres sportifs, ... etc.

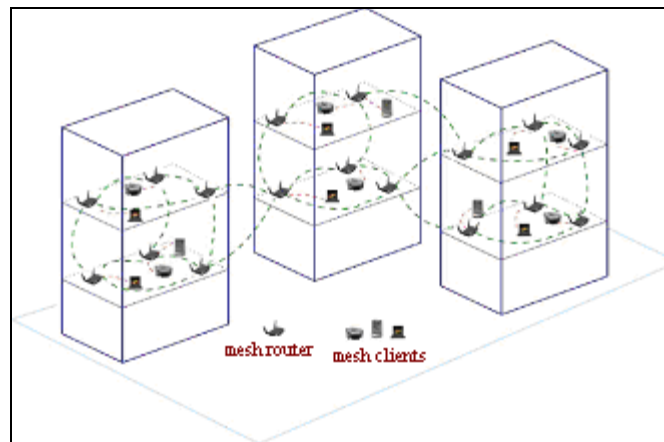


Figure 1.7 : Réseau d'entreprise

### 5. *Système de santé*

Dans un centre médical, les données de surveillance et de diagnostic doivent être traitées et transmises d'une chambre à une autre pour diverses fins. La transmission de ces données est à haut débit, car les images médicales de haute résolution et les diverses informations de contrôle périodiques peuvent facilement produire un grand volume de données. Les réseaux câblés traditionnels fournissent un accès limité à certains dispositifs médicaux fixes. Les réseaux Wi-Fi nécessitent l'existence de connexion Ethernet, qui coûte chère. Toutefois, cette contrainte ne se pose pas dans le cas de WMNs.

#### e) *Application de sauvetage*

Les réseaux maillés sans fil peuvent aussi être appliqués dans les applications spontanées (d'urgence/catastrophes). Par exemple, un réseau sans fil d'une équipe de réponse de secours où des pompiers n'ont pas de connaissance préalable où le réseau devrait être déployé. En plaçant simplement des routeurs maillés sans fil dans les endroits souhaités, un réseau maillé sans fil peut être rapidement établi. Pour un groupe de personnes détenant des appareils sans fil avec capacité d'être en réseau, par exemple, portables et PDAs, les communications P2P n'importe où et n'importe quand est une solution efficace pour partager des informations. Les réseaux maillés sans fil sont en mesure de répondre à cette demande. Ces applications illustrent que ce type de réseau est un ensemble de réseaux ad hoc, et peut donc accomplir toutes les fonctions prévues par un réseau ad hoc.

## 1.8 Conclusion

Le but de ce chapitre consiste à donner un aperçu général sur les réseaux maillés sans fil. Ces réseaux se caractérisent par la stationnarité des nœuds routeur, la mobilité des nœuds clients et la nature partagée du canal sans fil.

Ces caractéristiques exigent la définition de nouvelles métriques de routages pour l'établissement des routes entre les nœuds du réseau. Le chapitre suivant sera consacré à cette fin pour détailler ces spécificités et fera un tour d'horizon sur les protocoles de routage existants.

---

# Le routage dans les réseaux maillés sans fil

---

## 2.1 Introduction

D'après notre étude sur les protocoles de routage, nous pouvons classer les protocoles de routage développés pour les réseaux maillés sans fil selon le niveau de qualité de service QoS qu'ils fournissent en deux groupes: des protocoles de routage au mieux 'Best Effort' et des protocoles de routage conscients de la QoS. Ces protocoles de routage sont conçus à base de plusieurs métriques de routage que nous pouvons diviser en fonction du nombre de canaux utilisés dans la transmission de données en deux classes: métriques avec un seul canal et métriques avec plusieurs canaux.

Dans ce chapitre, nous analysons l'état de l'art des métriques de routage et nous proposons une taxonomie de protocoles de routage pour les réseaux maillés sans fil.

## 2.2 Métriques de routage

Le support de communication entre les nœuds du réseau exige l'utilisation des protocoles de routage qui doivent être combinés avec des métriques de routage pour déterminer le meilleur chemin entre une paire de nœuds. Cependant, la conception de métrique de routage efficace, dépend des caractéristiques spécifiques du réseau cible. Par exemple, les contraintes d'énergie de réseaux de capteurs exigent la conception d'un protocole de routage efficace en consommation d'énergie, tandis que la mobilité des nœuds dans les réseaux ad hoc exige la conception des protocoles qui peuvent maintenir efficacement la connectivité. La combinaison unique de nœuds statiques avec la nature partagée du moyen sans fil dans les réseaux maillés sans fil impose aussi des exigences spécifiques pour la conception de métriques de routage.

## 2.2.1 Métriques de routage avec un seul canal

### 2.2.1.1 Nombre de transmission attendu (ETX : Expected Transmission Count)

La métrique ETX [3] mesure le nombre moyen de transmissions nécessaires pour la bonne réception d'un paquet. Cette mesure peut être faite sur chaque lien du réseau. Ainsi, pour connaître le nombre moyen de transmissions nécessaires pour atteindre une destination, il suffit de faire la somme d'ETX sur chacun des liens traversés.

ETX est une mesure bi-directionnelle. Elle combine le *forward delivery ratio*, noté  $D_f$ , et le *reverse delivery ratio*, noté  $D_r$ .  $D_f$  est la probabilité qu'un paquet arrive avec succès au nœud suivant et  $D_r$  est la probabilité que le paquet ACK arrive lui aussi au nœud source. Une bonne transmission est l'association de ces deux étapes. ETX est donnée par la formule :

$$ETX(\text{link}) = 1 / (D_f * D_r) \quad (1)$$

Cette métrique se calcule par l'envoi périodique de paquets 'hello' aux nœuds voisins. Ces paquets sont comptabilisés et numérotés, ce qui permet d'en déduire aisément le nombre de transmissions réussies. Chacun des paquets envoyés contient ce nombre (calculé localement), ce qui permet à un nœud voisin de calculer la valeur d'ETX.

Plus la qualité de lien est mauvaise, plus la valeur de liaison d'ETX devient grande. La valeur d'ETX est égale à 1 pour une liaison parfaite qui ne perd aucun paquet. Si aucun paquet n'est reçu pendant une période prédéfinie, ETX est mis à 0 et on ne considère pas le lien pour le cheminement. Autrement, ETX est plus grand que 1.

Bien que ETX soit basé sur la probabilité de succès sur un seul lien, elle vise à réduire le nombre de transmissions au minimum tout au long d'un chemin donné, mais pas la probabilité de perte [4].

### 2.2.1.2 Perte minimale (ML : Minimum loss )

La métrique (ML) [4] est aussi basée sur l'envoi périodique des paquets 'hello' pour calculer le taux de livraison. ML trouve le chemin avec la plus basse probabilité de perte.

En interprétant les taux de livraison d'un lien comme étant des probabilités, la probabilité d'une transmission réussie de A à B est:

$$P_{AB} = D_f * D_r \quad (2)$$

Dans un chemin multi-sauts, la probabilité qu'une transmission réussie sur le chemin complet devrait être le produit des probabilités de chaque saut. Ainsi, dans un chemin de A à C, passant par B, on donne la probabilité totale d'une transmission réussie:

$$P_{AC} = P_{AB} \times P_{BC} \quad (3)$$

Les auteurs de ML argumentent que l'utilisation de multiplication réduit le nombre de changements de chemin, ce qui améliore les performances du réseau [3].

### 2.2.1.3 Temps de transmission attendu (ETT : Expected Transmission Time )

L'implémentation d'ETX a montré deux défauts [3]:

1. Les diffusions sont exécutées généralement dans un réseau de bas débit ;
2. Les paquets 'hello' sont plus petits que les paquets de données.

Ainsi, la performance d'ETX devient mauvaise parce que cette métrique ne prend pas en considération ni des liaisons avec différentes bandes passantes, ni la taille des paquets de données. Pour faire face à ces inconvénients, l'ETT d'un lien est défini comme étant un "ETX ajusté par la bande passante et la taille de paquet de données" Autrement dit, ETT est le temps nécessaire pour qu'un paquet de données soit transmis avec succès à chaque voisin.

Pour calculer ETT, on commence par le calcul de l'ETX et on le multiplie par la bande passante du lien pour obtenir le temps nécessaire pour la transmission d'un paquet. On peut le formaliser comme suit :

On note S la taille du paquet et B la bande passante du lien [5]. Alors on aura:

$$ETT = ETX * S/B \quad (4)$$

### 2.2.1.4 ETX modifié (mETX)

L'un des problèmes critiques des réseaux sans fil est la variation rapide de qualité de lien. Les métriques basées sur des valeurs moyennes calculées dans un intervalle de temps, comme ETX, ne peuvent pas suivre la variation de qualité du lien. Pour faire face à ce problème, deux nouvelles métriques mETX et ENT (Effective Number of Transmissions) ont été proposées [7]. Ces métriques envisagent l'écart-type en plus des valeurs moyennes de qualité du lien pour projeter les variations de la couche physique sur les métriques de routage.

La métrique mETX est aussi calculée en émettant des paquets 'hello'. La différence entre mETX et ETX est qu'au lieu de considérer des pertes de paquets, mETX travaille au niveau du bit. La métrique mETX calcule la probabilité d'erreur de bit en utilisant la position du bit corrompu dans le paquet et la dépendance de ces erreurs de bit au cours des transmissions successives. Cela est possible parce que les paquets sont composés par une séquence de bit connu [3].

### 2.2.1.5 Nombre Effectif de Transmission (ENT)

ENT est une approche alternative qui mesure le nombre de retransmissions successives par lien considérant la variation de qualité du lien. ENT émet aussi des paquets 'hello' et limite le calcul de chemin aux liens qui montrent un nombre acceptable de retransmissions selon des exigences de couche supérieure. Si un lien montre un certain nombre de transmissions attendues plus grand que le maximum toléré selon un protocole de couche supérieure (par exemple, TCP), ENT exclut ce lien du chemin et on lui associe une valeur infinie de métrique. mETX et ENT sont sensibles à la taille du paquet, donc l'inclusion de la bande passante est évidente avec ces deux métriques [3].

## 2.2.2 Métriques avec plusieurs canaux

### 2.2.2.1 Cumulatif pondéré d'ETT (WCETT : Weighted Cumulative ETT)

Les approches de 'Cross-layer' ont eu une attention spéciale dans les réseaux maillés sans fil [2]. Parmi les techniques disponibles, on trouve l'utilisation de plusieurs canaux.

Avec cette technique, il est possible d'améliorer les performances du réseau en utilisant, en même temps, les canaux non chevauchés disponibles définis par IEEE 802.11. Cependant, cette technique doit traiter deux problèmes pour devenir efficace, à savoir, l'interférence intra-flux et inter-flux.

- L'interférence 'intra-flux' arrive quand des nœuds différents transmettant des paquets du même flux s'interfèrent l'un avec l'autre. La minimisation du nombre de canaux n'est pas évidente, car les nœuds doivent maintenir leur connectivité.
- L'interférence d'inter-flux est autrement l'interférence subie entre des flux simultanés.

Pour réduire l'interférence intra-flux, WCETT [5] est proposé par Draves et d'autres pour réduire le nombre de nœuds qui utilisent un même canal pour la transmission d'un flux dans un chemin. Cette métrique est la combinaison entre le temps total de transmission de bout en bout et la diversité des canaux. Sa valeur est donnée par la formule suivante :

$$WCETT = (1 - \alpha) * \sum_{i=1}^n ETT_i + \alpha * \max_{1 \leq j \leq k} X_j \quad (5)$$

Où  $\alpha$  est un paramètre de combinaison compris entre 0 et 1,

$n$  : nombre de saut,

$k$  : nombre de canaux

$X_j$  est le temps de transmission écoulé en utilisant le canal  $j$  le long du chemin  $p$  et capture l'interférence intra-flow. Le composant de  $\max_{1 \leq j \leq k} X_j$  compte le temps maximal qu'un

même canal apparaît le long d'un chemin. Il capture l'interférence intra-flux d'un chemin puisqu'il donne essentiellement des poids bas aux chemins qui ont plus d'attributions de canal divers sur leurs liaisons.

$$X_j = \sum_{\text{Hop } i \text{ is on channel } j} ETT_i \quad 1 \leq j \leq k \quad (6)$$

Néanmoins, WCETT ne garantit pas de trouver le plus court chemin, et n'éviter pas les interférences inter-flux [6].

### 2.2.2.2 Interférence et commutation du canal

Cette métrique MIC [6] améliore WCETT, chaque nœud prend en compte le nombre de nœuds interférés dans le voisinage pour estimer l'interférence inter-flux. En outre, MIC utilise des nœuds virtuels afin de garantir le calcul des chemins du coût minimal. MIC calcule, également, sa valeur sur la base des métriques ETT.

### 2.2.2.3 Interférence (iAWARE)

Autre métrique qui considère aussi la variation de qualité de lien est iAWARE [8]. Cette métrique utilise SNR (Signal to Noise Ratio) et SINR (Signal to Interference and Noise Ratio) pour reproduire continuellement les variations d'interférence des voisins sur la métrique de routage. La métrique iAWARE estime le temps moyen que le canal soit occupé à cause des transmissions des voisins interférés.

Plus l'interférence est grande, plus la valeur d'iAWARE est grande. Ainsi, à la différence de mETX et ENT, iAWARE considère l'interférence intra et enter flux, l'instabilité du canal et le temps de transmission de données [3].

La plupart des implémentations des protocoles de routage au mieux préfèrent des métriques simples telles que ETX ou ETT. Le tableau 2.1 [3] résume les principales caractéristiques des métriques examinées.

Tableau 2.1 : Caractéristiques principales des métriques de routage

Métrique	Sensible au qualité	Bande passante	Taille de paquet	Interference intra_flux	Interference inter_flux	Instabilité du médium
Saut	×	×	×	×	×	×
ETX	√	×	×	×	×	×
ML	√	×	×	×	×	×
ETT	√	√	√	×	×	×
WCETT	√	√	√	√	×	×
MIC	√	√	√	√	√	×
mETX	√	√	√	×	×	√
ENT	√	√	√	×	×	√
iAWARE	√	√	√	√	√	√

### 2.3 Routage au mieux

Les services au mieux ‘Best effort’ sont ceux qui ne demandent pas des garanties de QoS. Un exemple de ce type de service est la navigation Web dans laquelle un utilisateur sollicite une page web Internet à travers un navigateur Web, puis attend sans aucune garantie que la page soit chargée sans un long retard. Cette page peut même ne pas être ouverte du tout en raison de la congestion du réseau ou de la surcharge du serveur. Pour ce type de trafic, le réseau essaie d’accommoder les flux autant que possible. Par conséquent, l’augmentation de la capacité du réseau devient la principale préoccupation.

D’après Yang et Al [6], les protocoles de routage dans les réseaux maillés sans fil peuvent être séparés en deux catégories, suivant la manière de création et de maintenance de routes lors de l’acheminement des données : les protocoles pro-actifs et les protocoles réactifs. Selon comment les paquets sont acheminés le long des chemins, les protocoles proactifs peuvent être encore divisés en deux sous-catégories : routage de source et routage saut par saut. Tous ces protocoles de routage ont des coûts différents en termes d’overhead de message et de complexité de gestion.

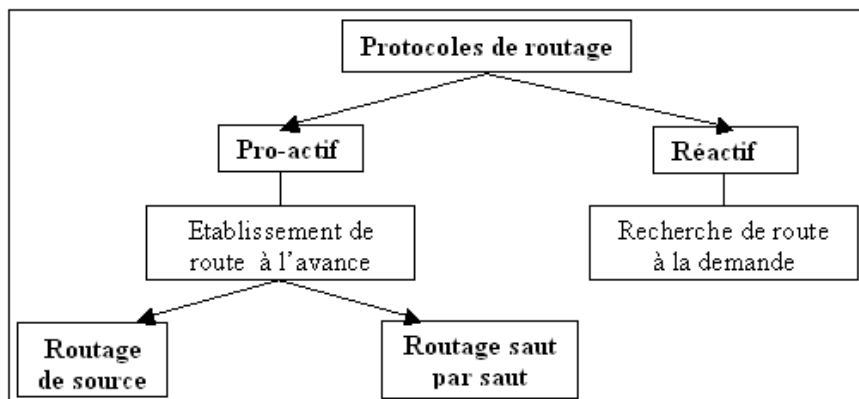


Figure 2.1 : Classification des protocoles

#### 2.3.1 Protocoles réactifs

Les protocoles de routage appartenant à cette catégorie, créent et maintiennent les routes selon les besoins. Lorsque le réseau a besoin d'une route, une procédure de découverte globale de routes est lancée, et cela dans le but d'obtenir une information spécifiée, inconnue au préalable. L'inondation de réseau est d'habitude utilisée pour découvrir les itinéraires qui sont nécessaires. Pour les réseaux ad hoc, puisqu'il y a des ruptures de lien fréquentes causées par la mobilité de nœuds, la découverte d'itinéraire à base d'inondation fournit la haute connectivité de réseau et les messages sont relativement de bas overhead comparés aux protocoles de routage proactifs. Cependant, dans les réseaux maillés, les liaisons ont d'habitude des durées de vie attendues beaucoup plus longues à cause de la nature statique des

nœuds. Puisque la fréquence de ruptures de lien est beaucoup plus inférieure que la fréquence de flux des arrivées dans des réseaux maillés, la découverte d'itinéraire à base d'inondation est considérée très coûteuse en termes d'overhead de message de contrôle. Donc, les protocoles réactifs ne sont pas généralement appropriés aux réseaux mesh.

Comme exemple de protocole de cette classe, nous citons AODV-ST [34] Ad hoc On-demand Distance Vector-Spanning Tree qui adapte le protocole AODV de réseaux ad hoc par l'inclusion de ETT comme métrique de routage. Dans ce protocole, les routeurs mesh construisent un arbre global (Spanning Tree) correspondant à chaque passerelle dans le réseau. Chaque routeur maintient une passerelle principale (qui a la plus petite valeur d'ETT) et achemine le trafic par cette passerelle. Les communications qui n'incluent pas la passerelle utilisent le protocole AODV original.

### **2.3.2 Protocoles pro-actifs**

Les protocoles de routage pro-actifs pour les réseaux maillés sans fil, sont basés sur la même philosophie des protocoles de routage utilisés dans les réseaux filaires conventionnels. Chaque nœud maintient un ou plusieurs tables de routage contenant des informations sur les autres nœuds du réseau. Quand des changements de topologie de réseau arrivent, les nœuds propagent des messages de mise à jour partout dans le réseau pour maintenir des informations de routage cohérentes et à jour sur le réseau, Ces protocoles de routage diffèrent dans la méthode par laquelle les paquets sont expédiés tout au long des chemins.

#### **2.3.2.1 Routage de Source**

Le routage de Source comme LQSR [3], impose la charge minimale en transmission de nœuds puisque le nœud source calcule le chemin pour un flux et le met entièrement dans les en-têtes de paquets. Des nœuds intermédiaires doivent seulement retransmettre les paquets en se basant sur les chemins trouvés dans les en-têtes de paquets. Cependant, en considérant que la taille de paquet dans les réseaux maillés est d'habitude très petite pour faire face au taux élevés d'erreur de bit de canaux sans fil et en mettant le chemin entier dans l'en-tête de paquet, cela impose un overhead de message.

Comme exemple de ce type de protocole, on propose le LQSR. Le protocole LQSR (Link Quality Source Routing) combine le protocole proactif d'état de lien avec la stratégie réactive de réseaux ad hoc. Comme un protocole d'état de lien, LQSR utilise une vue complète de la topologie de réseau pour calculer les chemins les plus courts. Néanmoins, LQSR utilise une procédure de découverte de chemin comme dans les protocoles réactifs pour réduire l'overhead, qui peut devenir grand à cause des instabilités des canaux sans fil et la mobilité des utilisateurs. Pendant la découverte de chemin, LQSR obtient la mise à jour des

informations d'état de lien exposé et des liens traversés, réduisant ainsi la périodicité de publicités régulières d'état de lien. LQSR utilise comme métrique ETX.

### **2.3.2.1 Routage saut par saut**

Dans le routage saut par saut, chaque nœud maintient une table de routage qui indique les sauts suivants pour les chemins à tous les autres nœuds dans le réseau. Pour qu'un paquet puisse atteindre sa destination, il doit connaître seulement son adresse. Les nœuds intermédiaires font suivre le paquet le long de son chemin en se basant seulement sur l'adresse de destination. En raison de sa manière d'expédition simple et de bas overhead de message, le routage saut par saut est dominant dans les réseaux câblés. Des raisons semblables font aussi que ce type de routage est le plus préférable pour les réseaux maillés sans fil.

Un exemple de ce type de protocole est le protocole HEAT [9], HEAT est inspiré du principe de champs de température dans la physique thermique. Un champ de température assigne une valeur scalaire à chaque particule dans l'espace. Les valeurs du champ de température sont plus hautes aux voisinages des sources de chaleur et ils diminuent en s'éloignant de la source. Par analogie, nous considérons les nœuds clients dans un réseau mesh comme des particules et les nœuds routeurs (passerelles) comme des sources de chaleur. Chaque nœud routeur initialise sa valeur de température à une valeur maximale, chaque nœud émet sa valeur de température à ses voisins périodiquement.

En se basant sur ces messages, tous les nœuds construisent et maintiennent une structure de données appelée table de voisinage, qui contient une entrée pour chaque voisin connu. Chaque entrée de la table contient l'adresse, la dernière température annoncée et une valeur d'horodatage du nœud voisin correspondant. Chaque fois qu'une entrée est ajoutée, enlevée, ou changée, la valeur de température d'un nœud est re-calculée par un algorithme qui prend en considération ses distances aux passerelles disponibles et la robustesse des chemins vers ces passerelles. Une fois que les champs sont calculés, l'acheminement des paquets vers les passerelles est mis en œuvre sur une base saut par saut : un paquet est toujours expédié au voisin qui a la plus haute température.

## **2.4 Routage avec Qualité de Service (QoS)**

Avec les avancées récentes dans la technologie de communication et l'inondation de services multimédia à travers Internet, la considération de QoS dans la gestion de réseau devient impérative. Fournir une meilleure QoS est un défi aux fournisseurs des services de réseau particulièrement dans l'environnement sans fil. Différentes couches peuvent être impliquées dans l'accomplissement des exigences de QoS individuelles ou intégrées à d'autres.

Les garanties de services sont une exigence typique pour des applications impliquant la transmission audio et vidéo comme la Voix sur IP (VOIP), l'enseignement à distance utilisant la transmission multimédia en ligne et la vidéo conférence. Ce type de trafic est caractérisé par le besoin de débit maximal. Dans ce cas, l'allocation de la bande passante est un défi ; si elle est allouée selon le débit maximal, des ressources sont gaspillées et d'autres applications seront bloquées tandis qu'il y a des capacités inutilisées. De plus, le calcul de la bande passante effective dans l'environnement radio n'est pas une tâche triviale.

Plusieurs métriques ont été utilisées pour spécifier les exigences de QoS dans les réseaux mesh (voir point 2.2). Une application peut exiger des garanties sur une ou plusieurs métriques pour son QoS. Dans ce qui suit, nous présentons et discutons les différentes approches de routage développées pour satisfaire les exigences de QoS.

### 2.5.1 Protocoles de routage Général

Un chemin est parcouru pour la satisfaction QoS dans une procédure semblable pour acheminer la découverte dans les protocoles de routage conventionnels. Le nœud source envoie un paquet de demande de chemin avec les garanties de QoS exigées sur le chemin comme la bande passante minimal exigée, le retard total attaché, le taux de perte de paquet ... etc. Cela peut être fait soit par l'inondation du réseau ou bien par l'utilisation d'un ou plusieurs chemins précédemment découverts. Chaque nœud recevant le RREQ (route request) portant les garanties de QoS exigées compare sa propre métrique mesurée avec la métrique exigée localisée dans l'en-tête du paquet. Si les exigences ne sont pas satisfaites, un paquet d'échec est renvoyé au nœud source pour commencer une nouvelle découverte de chemin ou à un nœud intermédiaire pour exécuter une procédure de réparation de chemin.

Dans [11] trois versions de protocole de routage sont présentées. Les protocoles sont conçus en supposant que les flux de données soient établis entre les nœuds clients et la passerelle. Deux nœuds clients communiqueront toujours par leur passerelle parentale. Donc, chaque nœud doit connaître un itinéraire à la passerelle la plus proche seulement.

Le protocole est nommé "Wireless Mesh Networks Routing Protocol" (MRP). La première version du protocole est MRP On demande (MRP-OD). Un nouveau nœud joignant le réseau diffuse localement (un saut) un message de découverte de route (route discover RDIS). Tous les nœuds recevant le message RDIS répondront avec un message route advertisement (RADV) contenant la métrique de leurs itinéraires à la passerelle la plus proche. Le protocole profite du fait que tous les nœuds connectés connaissent a priori des itinéraires aux passerelles et une métrique de cheminement significative de ces itinéraires. Si le nœud de

jointure n'a pas reçu de message RADV, il continuera à envoyer des messages RDIS. A la réception de messages RADV, le nœud sélectionnera un ou plusieurs itinéraires selon ses exigences QoS. De cette manière, la découverte de chemin amont du nœud à la passerelle est achevée. Le chemin de retour ne doit pas être le chemin inverse. Ainsi, une découverte de chemin de retour est établie en envoyant un message (Route REGistration) RREG du nœud à la passerelle. Quand la passerelle reçoit le message RREG, elle répondra par un message route acknowledgement (RACK). La perte de n'importe quel RREG ou RACK exige le recommencement du processus de découverte. A la réception du RACK, le nœud de jointure commencera à envoyer et recevoir de l'Internet.

Si une perte de lien est soupçonnée pendant la transmission et la réception de paquets de données, le nœud entre dans un état de « vérifier-lien ». Dans cet état, le nœud continue à envoyer des paquets de route check (RCHK) au saut suivant. Si une réponse est reçue alors le chemin sera utilisé de nouveau, autrement le nœud se commute à un état « débranché ». Si un nœud perd son chemin vers la passerelle, tous ses enfants perdront leur chemin également, ainsi quand un nœud perd sa connexion avec la passerelle, il envoie un message route error (RERR) à tous ses enfants. En perdant la connexion à une passerelle, un nœud et tous ses enfants, réinitialisent un processus de découverte de chemin.

Le diagramme suivant, montre le principe de fonctionnement de ce protocole :

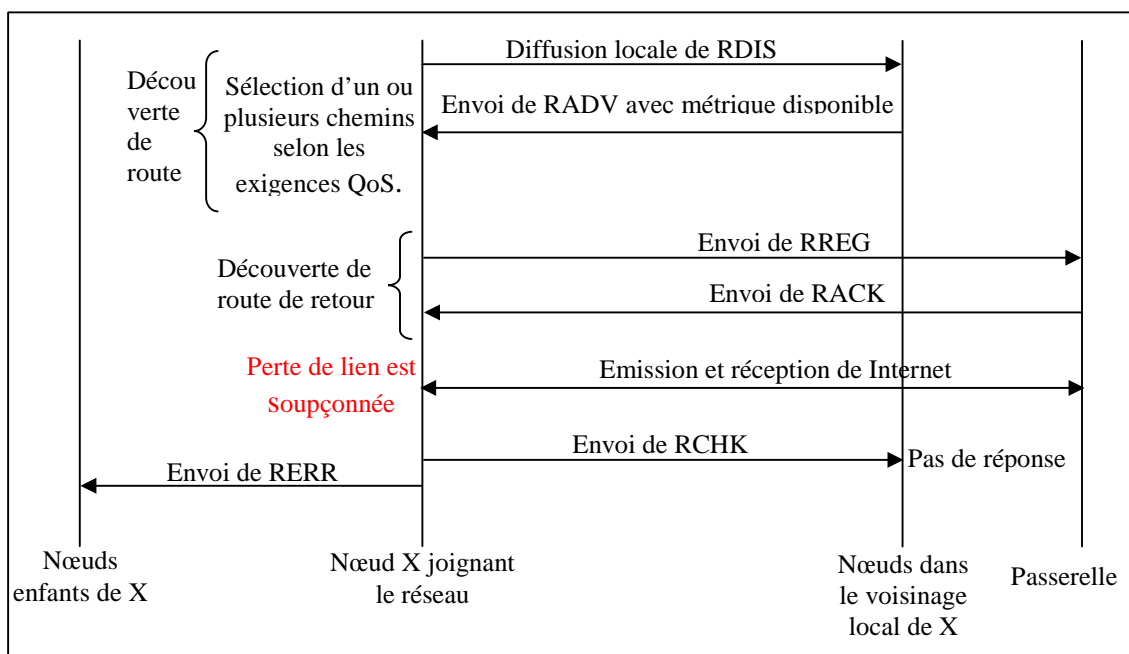


Figure 2.2 : Principe de fonctionnement de MPR-OD

La deuxième version de MRP est le mode beacon MRB (MRP B). Au lieu d'envoyer des messages RDIS aux voisins pour découvrir un itinéraire à une passerelle, un nœud écoute des

beacons périodiques envoyés par des nœuds dans le réseau pour faire de la publicité pour des itinéraires. Après la collecte des beacons à partir des voisins, la sélection d'itinéraire et l'enregistrement continuent comme dans MRP-OD. Une perte de liaison est décidée en perdant un nombre prédéterminé de beacons. Cette version réduit le retard de détection de route endommagé, mais augmente l'overhead de routage.

La troisième version est MRP hybride (MRP H) qui est une combinaison des deux versions précédentes. Un nouveau nœud joignant le réseau envoie des messages RDIS pour découvrir des itinéraires à la passerelle au lieu d'attendre les beacons tandis qu'une perte de lien est détectée en ne recevant pas un nombre prédéterminé de beacons. Cette version a une découverte d'itinéraire et une détection d'échec de liaison plus rapides. Cependant, elle impose plus d'overhead de routage par rapport aux deux autres versions.

### 2.5.2 Routage à base de localisation

Pour les protocoles de routage généraux, un protocole de routage doit inonder le réseau avec des paquets de RREQ pour découvrir un chemin vers la destination. Cet overhead peut être énormément réduit si le nœud a des informations sur la localisation de destination. Dans les protocoles de routages basés sur la localisation, le nœud source envoie un paquet avec la position de destination contenue dans son en-tête aux voisins les plus proches de la destination. Les voisins font suivre le paquet aux voisins suivants les plus proches à la destination et ainsi de suite jusqu'à ce que le paquet atteigne la destination. Cette dernière peut être un nœud seul ou un groupe de nœuds dans un certain secteur géographique qui est appelé geocasting. Les informations de localisation peuvent être obtenues en utilisant le système GPS ou d'autres dispositifs de détermination de position [12]. Un protocole récent nommé QoS-GPSR (Greedy Perimeter Stateless Routing) [13] a utilisé des technologies Ultra WideBand (UWB) pour estimer la position. Dans ce protocole, le nœud source envoie un RREQ pour découvrir un chemin qui satisfait les exigences de QoS. Les paquets de données sont envoyés à la destination selon la direction estimée tant que les exigences QoS sont satisfaites. Si les exigences QoS sont violées, un paquet d'échec d'itinéraire est envoyé au nœud source pour commencer une autre découverte de route avec QoS.

### 2.5.3 Routage avec Différenciation de Service

Dans le contexte de l'Internet, la proposition *DiffServ* prône pour une réservation de ressources sur un ensemble de flots pour fournir de la qualité de service (QoS). La QoS peut être fournie, soit au niveau MAC, soit au niveau IP. La première stratégie définit la politique

pour le choix du nœud qui accédera au support tandis que la seconde concerne le choix du paquet à transmettre au sein d'un nœud [14].

Au niveau MAC, l'idée est de donner une priorité d'accès au support pour les paquets des flots à service privilégié. Au niveau IP, le problème que posent les différentes approches concerne la sous-utilisation des ressources. Il existe peu de propositions qui réalisent une redistribution dynamique des ressources entre les nœuds. La gestion de la QoS au niveau IP permet de contrôler le trafic émis dans chaque classe de service.

DiffServ définit 3 types de classes de services [15] qui sont :

- **EF (Expedited Forwarding)**

C'est la classe d'excellence. Les paquets marqués EF doivent être acheminés avec un délai, une gigue (Variation du délai de transfert de l'information, due principalement aux délais de transferts variables dans les nœuds du réseau) et un taux de perte minimum. Cette classe est destinée aux applications en temps réel comme la voix sur IP (VoIP), etc.

- **AF (Assured Forwarding)**

Cette classe est appropriée aux utilisateurs exigeant des services fiables avec un taux cible nommé committed information rate (CIR).

Deux classes AF ont été définies, AF\_in et AF\_out. Si le taux de transmission mesuré d'un client de cette classe n'excède pas son CIR, tous les paquets de ce client sont marqués dans la sous-classe AF\_in; autrement, les paquets seront désignés dans la sous-classe AF\_out. En cas de congestion, les paquets de la classe AF\_out verront une probabilité plus haute d'être choisis que les paquets de la classe AF\_in, réalisant ainsi la différenciation de service par flux.

- **BE (Best Effort)**

C'est l'équivalent de l'Internet actuel où aucun traitement particulier ne vient améliorer le relayage des paquets appartenant à cette classe.

Une architecture pour la disposition de services différenciée dans le backbone maillé sans fil est discutée dans [15]. Hai et Al ont fait une étude sur la différenciation de service dans les réseaux sans fil et une comparaison avec les réseaux filaires. Cette étude montre que les nœuds du réseau sans fil doivent jouer les deux rôles définis pour les nœuds d'un réseau filaire. Ainsi un nœud sans fil est considéré comme un nœud de bord qui assure les fonctions de classification des paquets en un nombre limité de classes de service, selon le Service Level Agreement (SLA) négocié avec le Fournisseur de service Internet (ISF). En même temps, il joue le rôle d'un routeur de cœur qui garantit la différenciation de l'acheminement des paquets provenant de différentes classes par différents comportements pour chaque saut (PHBs : per-hop behaviors). Les routeurs sans fil utilisent alors plusieurs files d'attente

séparées, contrôlées par certains algorithmes d'ordonnement pour présenter les classes différenciées dans le backbone maille sans fil.

Hai et Al ont proposé par la suite un routage de QoS et un plan de MAC pour les routeurs de backbone sans fil. Ils recommandent que le routage soit à base de classes car pour chaque classe, le protocole de routage devrait choisir les chemins qui satisfassent les exigences QoS de cette classe. Ils supposent que le réseau travaille avec un seul canal. Ce travail est préliminaire et reste ouvert à la recherche.

#### **2.5.4 Routage avec contraintes multiples**

Les flux de trafic varient dans leurs exigences. Les flux peuvent être sensibles à la bande passante (exige une bande passante minimale), ou sensibles au retard (un retard maximal attaché). D'autres peuvent exiger un taux de perte de paquet maximal (PLR). Les flux demandant des garanties de métrique QoS multiples imposent plus de difficulté. La résolution d'un problème de chemin multi-contraintes est un problème NP-complet [16] dans des nombreux cas comme celui dans lequel la métrique est additive. Même la satisfaction d'une contrainte peut devenir NP-complet quand l'interférence de liaison est prise en considération.

La résolution du problème de routages multi-contraints de WMNs est dressée dans [16]. Les auteurs proposent un protocole nommé 'QUORUM' qui exécute sur demande la découverte et la sélection d'itinéraires en utilisant plusieurs métriques comme la bande passante, le retard et la robustesse et en fournissant des garanties de QoS.

Le protocole dans [13] utilise des heuristiques pour résoudre le problème. Les chercheurs divisent le problème à deux contraintes dans deux problèmes à une contrainte. Une découverte de chemin de QoS est faite en deux étapes. La première doit découvrir un chemin qui satisfait l'exigence de taux de perte de paquet. La deuxième étape doit vérifier si le chemin découvert satisfait l'autre exigence qui est la bande passante ou le retard.

#### **2.5.5 Routage avec canaux multiples**

Le Multi-canaux est présenté comme une solution pour le problème d'interférence. Les Canaux peuvent être des codes orthogonaux (CDMA) ou des bandes de fréquences (FDMA). Un protocole de routage qui est construit sur un réseau multi-canaux est surtout articulé avec le problème d'attribution de canal. Ce problème a été examiné dans plusieurs travaux [12].

Un protocole de routage sensible à la bande passante (BAR:Bandwidth Aware Routing) est présentée dans [17]. Le protocole de routage garantit une bande passante pour chaque flux en

supposant qu'on connaît la bande passante pour chaque lien. On suppose que chaque nœud est équipé de deux cartes d'interface de réseau (NICs). Chaque NIC est accordé à un canal différent. Les différents nœuds utilisent des canaux différents avec la contrainte que chaque couple de nœuds voisins doit avoir un canal commun pour communiquer. L'attribution de canal est centralisée et statique, ce qui signifie que cela durera tant qu'il n'y a aucun nouveau nœud supplémentaire dans le réseau [12].

### **2.5.6 Routage avec multi-chemin**

Le routage avec Multi-chemin est utilisé pour maximiser l'utilisation des ressources du réseau. Avec le routage multi-chemin, un nœud choisit plusieurs liens pour la même destination. S'il est correctement géré, il peut agréger les ressources de chemins multiples, augmenter le taux d'admission et améliorer la fiabilité.

La conception d'un protocole de multi-chemin exige beaucoup de soin sur les frais de cheminement et la coordination avec les flux voisins. Le protocole proposé dans [18] limite le nombre de chemins multiples à un nombre indiqué par le flux. Le protocole, nommé multi-path dynamic source routing protocol (MP-DSR), est basé sur le protocole Dynamic Source Routing existant (DSR) et profite de sa nature on-demand distribué. Il cherche à calculer un ensemble de chemins disjoints qui peuvent satisfaire une exigence de fiabilité continue minimale.

La fiabilité de chemin est calculée en se basant sur la disponibilité de tous les liens le long d'un chemin. La disponibilité de lien est définie comme la probabilité qu'un lien est disponible pendant une période définie de temps. Le calcul de disponibilité de lien est basé sur le mouvement du nœud; une contrainte qui n'est pas trouvée dans les réseaux maillés sans fil.

La même approche de limitation de chemins est dressée dans [19]. Chen propose 'a ticket based multipath routing'. Le nœud source publie des sondes pour découvrir des chemins. Chaque sonde est assignée à un certain nombre de tickets qui sont égaux au nombre de chemins multiples exigés. Le nombre de tickets est rapproché de la rigueur des exigences de contrainte.

### **2.5.7 Routage avec puissance de transmission**

Comme la contrainte de consommation d'énergie n'existe pas dans les réseaux mesh, la puissance de transmission devient un facteur important dans la conception d'un protocole de routage. Tamer dans [12] a proposé un protocole de routage avec puissance de transmission

qui garantit trois exigences de QoS à savoir : le retard, le bruit du signal et le taux d'interférence.

Ses résultats montrent que les chemins ont tendance à avoir moins de sauts de la source à la destination pour maximiser le débit du réseau. Tandis qu'en cas d'augmentation des flux du réseau, les chemins ont tendance à avoir plus de sauts avec des longueurs plus courtes pour réduire l'interférence et tenir compte de la réutilisation de fréquence. On augmente la réutilisation de fréquence pour augmenter le nombre de flux dans le réseau jusqu'à un certain point où le réseau est saturé et aucun espace pour de nouvelles liaisons qui permettent de travailler simultanément n'est disponible.

Ce travail utilise un contrôleur centralisé que l'on considère être un responsable de groupe dans un réseau groupé. Une extension importante à ce travail doit trouver une mise en œuvre distribuée. De plus la scalabilité du réseau vaut la peine d'être dressée. On remarque qu'en augmentant la taille de réseau et le nombre de flux dans le réseau, les calculs deviennent plus complexes et exigent une longue période de temps. Dans ce cas, une heuristique pour réduire la complexité est exigée. L'heuristique peut diminuer l'exactitude des résultats, qui est acceptable pour permettre l'évaluation de grands réseaux.

### 2.5.8 Synthèse

Nous avons résumé dans le tableau suivant les différentes approches étudiées. Nous avons identifié pour chaque approche : son principe, un exemple de protocole réalisé et les problèmes rencontrés.

**Tableau 2.2 : Approche de routage avec QoS dans les réseaux mesh**

Approches	Principe	protocole	Problèmes
<b>Générale</b>	Inonde le réseau par des RREQ avec la métrique exigée	MRP	overhead
<b>Localisation</b>	Reduit l'overhead dû à l'inondation et utilisation des informations de localisation	QoS-GPRS	Dépend d'un système de localisation
<b>Différentiation de service</b>	MAC : choisit le nœud qui accède au canal (par priorité), Routage : choisit le paquet à transmettre au sein d'un nœud		Distribution dynamique des ressources entre les nœuds
<b>Multi contraintes</b>	Satisfait plusieurs contraintes	QUORUM	Problème NP complet (pas de solution exacte)
<b>Multi chemin</b>	Maximise l'utilisation des ressources du réseau.	MP-DSR	overhead
<b>Multi canaux</b>	Résout le problème d'interférence	BAR	Attribution de canal
<b>Puissance de transmission</b>	Fait varier la puissance de transmission pour réduire l'interférence		Non scalable

## 1.5 Conclusion

Dans ce chapitre, nous avons cité les différentes métriques de routages proposées pour les réseaux maillés sans fil. Les protocoles de routage ont été classifiés selon le niveau de QoS qu'ils fournissent en deux groupes : routage au mieux 'best effort' et routage avec QoS. Nous avons examiné les différentes classes de protocoles de routage au mieux et nous avons trouvé que le routage proactif saut par saut est le type le plus approprié pour les réseaux maillés sans fil. Par la suite, nous avons étudié les différentes approches de routages avec QoS proposés pour ce type de réseaux.

Nous avons conclu que peu de travaux ont été faits sur l'approche multi-canaux et celle de différenciation de service qui sont deux approches très prometteuses pour le soutien de QoS. Leur combinaison peut faire l'objet d'un bon protocole de routage de QoS. Pour cela, nous allons consacrer le prochain chapitre pour bien étudier le problème d'attribution de canal qui est le problème majeur dans l'approche multi-canaux.

---

## Attribution de canal

---

### 3.1 Introduction

En raison de la capacité de canal sans fil limitée, de l'influence d'interférences, du grand nombre d'utilisateurs et de l'apparition d'applications multimédia en temps réel, l'amélioration du débit du réseau et le support de QoS sont devenus deux exigences critiques dans de tels réseaux.

L'utilisation de canaux multiples au lieu d'un seul canal dans les réseaux sans fil multi-sauts peut améliorer significativement le débit du réseau. Les normes IEEE 802.11b et IEEE 802.11a offrent respectivement 3 et 12 canaux non chevauchés. L'amélioration du débit de tels réseaux est apportée principalement en permettant des transmissions simultanées multiples dans un voisinage.

L'architecture d'un réseau maillé sans fil multi-radio pose deux problèmes:

1. Lequel des 3 ou 12 canaux sans fil devrait être assigné à une interface donnée?

Pour que deux noeuds puissent communiquer entre eux, leurs interfaces doivent être assignées à un canal commun. Cependant, quand plusieurs interfaces dans une gamme d'interférences sont assignées à un même canal sans fil, la bande passante efficace disponible pour chaque interface diminue. Donc, un algorithme d'attribution de canal doit équilibrer entre les buts de maintenir la connectivité du réseau et augmenter la bande passante globale.

2. Comment les paquets devraient être acheminés dans ce réseau multi interface?

La stratégie de routage dans le réseau détermine la charge de trafic sur chaque interface et affecte à son tour l'exigence de la bande passante et ainsi l'attribution de canal de chaque interface.

Dans ce chapitre, nous donnons un état de l'art sur les algorithmes d'attribution de canaux dans les réseaux maillés sans fil multi-canaux multi-radio. Pour cela, nous citons dans un

premier temps les ressources à gérer dans ce type de réseau, et dans second temps, nous discutons les différences et les défis de l'attribution de canaux dans les réseaux mesh sans fil comparés aux réseaux cellulaires. Dans un troisième temps, nous donnons quelques notions nécessaires; et nous mettons en évidence les contraintes associées et les défis d'attribution de canaux dans les réseaux maillés sans fil. Nous classifions, dans un quatrième temps, les algorithmes d'attribution de canaux selon plusieurs critères et nous présentons les travaux faits dans ce domaine. Nous terminerons enfin par une synthèse et une conclusion.

### **3.2 Gestion des ressources dans un réseau maillés sans fil multi-radio multi-canaux**

D'après [20] "Une ressource est n'importe quelle chose qui est exigée pour exécuter une tâche et qui est consommée pendant l'exécution". Par analogie, les ressources à gérer dans les réseaux maillés sans fil Multi-Canaux Multi-Radio (MC-MR) sont les interfaces radio et les canaux orthogonaux.

Le problème de gestion de ces ressources est particulièrement stimulant car il exige la solution des problèmes d'optimisation où beaucoup de variables d'assignation couplées doivent être conjointement optimisées [21].

- Le choix des interfaces pour les communications,
- L'attribution des canaux aux interfaces,
- La planification de leurs transmissions et
- Le cheminement du trafic sur la topologie résultante

L'attribution du canal est la tâche fondamentale d'un gestionnaire de ressource, car elle influence sur l'augmentation de la bande passante et la qualité de service de telle sorte qu'une utilisation efficace des fréquences soit possible pour éliminer le problème d'interférence [22].

### **3.3 Attribution du canal dans les Réseaux Cellulaires / Réseaux maillés sans fil**

Le problème d'attribution de canal a été largement étudié dans le contexte des réseaux cellulaires sans fil [36]. Le concept de base utilisé est la division du spectre radio en un ensemble de canaux disjoints non-interférant. Ces canaux peuvent être alors utilisés simultanément en maintenant une séparation de canal adjacente.

Différentes techniques ont été utilisées pour diviser le spectre radio, tel que, division de fréquence (FD), division de temps (TD) ou division de code (CD). Dans FD, le spectre est divisé dans des bandes de fréquences disjointes. Tandis que dans TD, la séparation de canal est réalisée en divisant l'utilisation de canal dans des fentes de temps, une combinaison de FD et TD peut aussi être utilisée pour diviser chaque bande de fréquences en fentes de temps.

Soit  $S_I(k)$  est l'ensemble I de terminaux sans fil, qui communiquent avec la station de base utilisant le canal k. À cause de la pénurie du spectre radio, il y a un nombre limité de canaux. Ainsi le même canal k peut être réutilisé simultanément par un autre ensemble J si les membres des ensembles I et J sont assez espacés. Ces ensembles, utilisant le même canal, sont appelés des co-canaux. Le concept de réutilisation de canal est illustré dans la figure 3.1, où il y a sept canaux orthogonaux disponibles (étiqueté de A à G). Chaque canal est utilisé pour la communication à l'intérieur d'une cellule et est réutilisé simultanément par une autre cellule qui est assez loin.

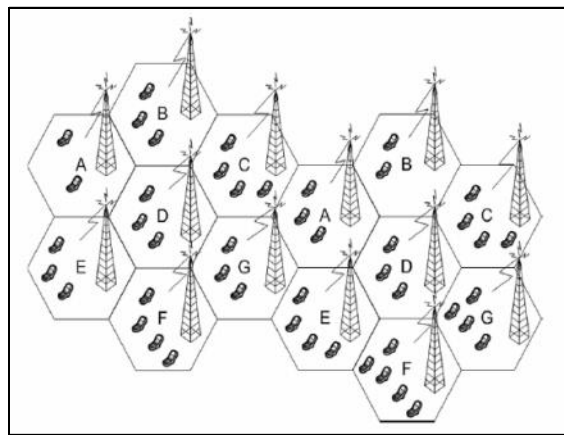


Figure 3.1 : Réutilisation de canal dans les réseaux cellulaires

La distance minimale à laquelle les co-canaux peuvent être réutilisés avec interférence acceptable est appelée la distance de réutilisation de co-canal. C'est possible parce qu'en raison de la perte de chemin, la puissance moyenne reçue d'un émetteur à la distance  $d$  est proportionnelle à  $P_T d^{-\alpha}$ , où  $\alpha$  est dans la gamme 3-5 selon l'environnement physique et  $P_T$  est la puissance de transmission moyenne. L'interférence de co-canal causée par la réutilisation de fréquence est un facteur de restriction sur la capacité du système. Le rôle d'un protocole d'attribution de canal est donc de réduire au minimum cette interférence en ajustant (1) la distance entre les co-canaux et/ou (2) le niveau de puissance de transmission. Ces deux méthodes (1 et 2) présentent le concept sous-jacent pour l'attribution de canal dans les réseaux cellulaires dont le but est de réduire au minimum le taux CIR (carrier-to-interference ratio) et augmenter l'efficacité de réutilisation de spectre radio.

Par contraste avec cela, le problème d'attribution de canal dans les réseaux maillés sans fil est différent en termes de plusieurs aspects.

- Tout d'abord, l'architecture d'un réseau maillé sans fil diffère de celle d'un réseau cellulaire. Dans un réseau maillé sans fil, les nœuds routeurs forment un backbone radio multi-sauts entre les nœuds clients et le réseau câblé, alors que dans un réseau cellulaire,

les terminaux des utilisateurs finaux communiquent directement par un seul saut avec la station de base, et la communication entre les stations de base est portée sur un réseau séparé qui n'intéresse pas au problème d'attribution de canal.

- Deuxièmement, l'attribution de canal dans les réseaux maillés sans fil est destinée principalement pour minimiser l'interférence dans le backbone sans fil. Le backbone est le centre principal de recherche dans l'amélioration de capacité des réseaux maillés sans fil. L'attribution du canal dans les réseaux cellulaires est seulement concernée par la minimisation de l'interférence sur le dernier saut de communication entre la station de base et les dispositifs de portable d'utilisateur final et vice versa.
- De plus, FH (frequency hopping) est une technique généralement utilisée dans les réseaux cellulaires et consiste à commuter rapidement les fréquences pendant la transmission sans fil par la station de base. FH a beaucoup d'avantages, particulièrement dans la réduction de l'effet de bruit et de l'interférence. Cette technique pourrait probablement être utilisée dans les réseaux maillés sans fil, Cependant avec le matériel actuel de la norme IEEE 802.11, la latence de temps de commutation est toujours extrêmement haut [26] (par exemple dans l'ordre de millisecondes). Une telle commutation de canal est donc difficile et cela rend l'attribution de canal dans les réseaux maillés sans fil plus stimulante.

### 3.4 Préliminaires

Avant que nous présentions une taxonomie des stratégies d'attribution de canal existantes dans les réseaux maillés sans fil, nous fournissons d'abord quelques concepts et définitions de base.

#### 3.4.1 Le graphe de Connectivité

Nous considérerons un réseau maillé sans fil avec des nœuds routeurs distribués sur un terrain. Chaque nœud routeur est équipé par un ou plusieurs interfaces radios avec des antennes omnidirectionnelles. Nous supposons que tous les canaux sont caractérisés par une gamme de transmission identique ( $R$ ) et aussi par la même gamme d'interférence ( $R'$ ). La gamme de transmission est définie comme la distance à laquelle un voisin peut recevoir des transmissions de paquet avec succès. Quand un récepteur est dans la gamme de transmission de deux émetteurs qui transmettent simultanément, les paquets s'interfèrent entre eux. Cela mène alors à une collision au récepteur et ainsi aucun paquet ne sera reçu avec succès. La gamme d'interférence est définie comme la distance à laquelle la transmission de paquet ne peut pas être décodé avec succès par le récepteur. Cependant, n'importe quelle nouvelle transmission d'un routeur dans la gamme d'interférence du récepteur interfère avec la

réception du paquet. Il est généralement assumé que la gamme de transmission est plus petite que la gamme d'interférence ( $R < R'$ ) [37].

Selon les susdites suppositions, la connectivité entre les nœuds routeurs peut être modélisée en utilisant un graphe non orienté appelé graphe de connectivité  $G$ . Comme illustrés dans la figure 3.2, deux nœuds dans le graphe de connectivité sont liés s'ils sont placés dans la gamme de transmission l'un de l'autre (voir le modèle de protocole, expliqué dans le paragraphe suivant). En général, la topologie de réseau (aussi appelé la topologie logique) diffère du graphe de connectivité,

- a) une liaison dans le graphe de connectivité peut être absente dans le graphe de topologie de réseau si les nœuds constituant cette liaison n'ont pas d'interfaces radios assignées à un canal commun; et
- b) une liaison dans le graphe de connectivité peut avoir plusieurs liaisons correspondantes dans le graphe de topologie de réseau si les nœuds ont plus qu'une interface radio chacun avec des canaux communs. Notez que les liaisons présentes dans la topologie de réseau sont appelées liaisons logiques.

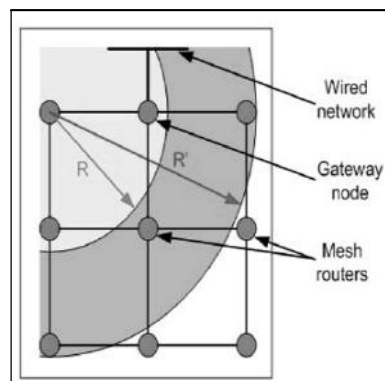


Figure 3.2 : Exemple de graphe de connectivité

### 3.4.2 Le graphe de Conflit

A cause de la nature de diffusion du moyen sans fil, le succès de transmission est énormément influencé par la quantité des accès multiples interférant. Cette interférence peut être modélisé en utilisant un graphe de conflit tiré à partir du graphe de connectivité. Le concept d'un graphe de conflit est illustré dans la figure 3.3, où une liaison entre deux nœuds  $x$  et  $y$  dans le graphe de connectivité de la figure 3.3 (a) est représentée par un sommet  $l_{xy}$  Dans le graphe de conflit de la figure 3.3 (b). Nous utilisons les termes "nœud" et "liaison" en ce qui concerne le graphe de connectivité et réservons les termes "sommet" et "arc" pour le graphe de conflit, comme dans [37]. Un arc est placé entre deux sommets dans le graphe de conflit si les liaisons

correspondantes dans le graphe de connectivité s'interfèrent. L'existence et la mesure d'interférence entre une paire de liaisons sont décidées par un modèle d'interférence.

Il y a deux modèles d'interférence bien connus : (i) le modèle de protocole et (ii) le modèle physique. Le modèle de protocole est le plus simple et le plus généralement utilisé pour représenter l'interférence tandis que le modèle physique est plus complexe, mais il offre un paradigme plus réaliste. En assumant que tous les nœuds dans le réseau ont la même gamme d'interférence, la transmission de  $x$  à  $y$  n'est réussie que si aucun autre nœud situé dans la distance  $R'$  de  $y$  transmet en même temps que  $x$ . De plus, dans le cas de protocole IEEE 802.11, Si le mode RTS/CTS (Request to Send/Clear to Send) est utilisé, donc aucun autre nœud dans la distance  $R'$  de  $x$  ne devrait transmettre en même temps. Le graphe de conflit pour le modèle de protocole contient donc un arc entre deux sommets (c'est-à-dire  $I_{xy}$ ,  $I_{xz}$ ) si  $x$  ou  $y$  sont placés dans la distance  $R'$  de  $z$ .

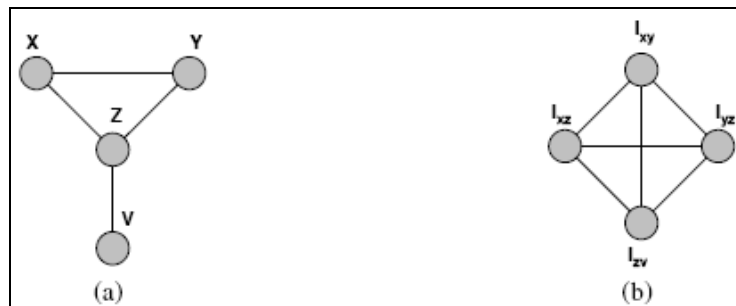


Figure 3.3 : Exemple de graphe de conflit

Parallèlement, dans le modèle d'interférence physique, supposons que le nœud  $x$  veut transmettre au nœud  $y$ . La puissance de signal  $SS_{xy}$  de la transmission de  $x$  est calculée comme reçue à  $y$ . La transmission est réussie si  $SNR_{xy} \geq SNR_{resh}$ , où  $SNR_{xy}$  est le rapport entre le signal et le bruit dans  $y$  de la transmission reçue de  $x$  ( $SNR$  : the signal to noise ratio). Le bruit total  $N_y$  dans  $y$  est le bruit total des voisins ( $N_a$ ) et l'interférence causée par d'autres transmissions en cours dans le réseau. Basé sur ce modèle, une liaison  $I_{xy}$  existe entre  $x$  et  $y$  dans le graphe de connectivité si et seulement si  $SS_{XY}/N_a \geq SNR_{resh}$  (c'est-à-dire le  $SNR$  excède au moins le seuil minimal en présence de bruit voisin seulement). L'interférence dans le modèle physique augmente progressivement quand des nœuds plus voisins transmettent et devient inacceptable quand le niveau de bruit atteint le seuil. Cette augmentation graduelle implique que le graphe de conflit devrait être un graphe pondéré, où le poids d'un arc orienté entre deux sommets indique la fraction du bruit toléré au nœud de réception [37].

### 3.4.3 Le graphe de Conflit Multi-radio

Le graphe de conflit multi-radio (MCG) [28] est une extension du graphe de conflit décrit dans le paragraphe précédent. Dans le MCG, au lieu de représenter les liaisons entre des nœuds routeurs, les sommets représentent les liaisons entre les interfaces radios. Pour créer le MCG, chaque interface radio dans le réseau est représentée par un nœud dans un nouveau graphe  $G'$  au lieu de représenter des routeurs par des nœuds comme dans  $G$ .

Dans l'exemple précédent, supposons que le nœud  $z$  a deux radios et le reste des nœuds ont une seule interface radio comme indiqué dans la figure 3.4 (a). Le nœud  $z$  sera représenté donc par deux nœuds dans  $G'$  comme dans la figure 3.4 (b), correspondant à ses deux interfaces radios, au lieu d'un seul nœud seulement comme dans  $G$ . Alors chaque liaison dans  $G'$  est représentée en utilisant un sommet dans le MCG

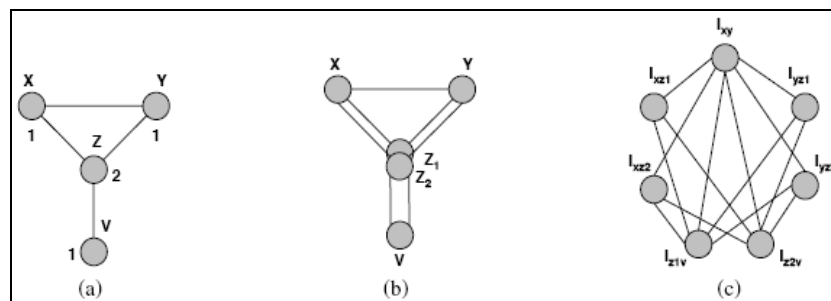


Figure 3.4 : Exemple de graphe de conflit multi-radio

Les arcs entre les sommets dans le MCG sont créés par la même façon que dans le graphe de conflit original. Deux sommets dans le MCG ont un arc entre eux si les liaisons dans  $G'$  représentées par ces deux sommets s'interfèrent. La figure 3.4 (c) montre le MCG du réseau maillé sans fil représenté dans la figure 3.4 (a). Dans cette figure, on étiquette chaque sommet en utilisant les interfaces radios qui composent le sommet. Par exemple, le sommet  $xz2$  représente la liaison entre l'interface radio sur le routeur  $x$  et la deuxième interface radio sur le routeur  $z$ .

### 3.5 Les contraintes et les défis dans l'attribution de canal

Prenons le graphe de connectivité et le modèle d'interférence, le défi principal pour l'attribution de canal est de savoir comment assigner un canal à chaque interface radio de telle sorte qu'on réduise au minimum l'interférence et on maximise la connectivité entre les nœuds. Les contraintes principales [25] qu'un algorithme d'attribution de canal devrait satisfaire sont :

1. Le nombre total de canaux est fixé.
2. Le nombre de canaux distincts qui peuvent être assignés à un nœud routeur est limité par le nombre de ses interfaces radios.
3. Deux nœuds qui partagent une liaison virtuelle devraient nécessairement avoir un canal commun.
4. La somme de charge du trafic attendu sur les liaisons qui partagent un même canal et qui s'interfèrent entre eux ne devrait pas excéder la capacité brute du canal.

À première vue, l'attribution de canal semble être un problème direct de coloriage de graphe [25]. Cependant, le problème de coloriage de graphe standard ne peut pas capturer les susdites contraintes et les spécifications du problème. Une formulation "node-multi-coloring" [37] échoue à capturer la troisième contrainte où les nœuds communicants ont besoin d'une couleur commune. Parallèlement, une formulation "edge-coloring" échoue à capturer la deuxième contrainte où pas plus que le nombre d'interfaces radios d'un nœud couleurs peut être donné à un nœud. Bien que la contrainte de coloriage d'arc puisse être grossièrement capable de modéliser les contraintes restantes, celle-ci ne peut pas satisfaire la quatrième contrainte de capacité de canal limitée.

De plus, un problème clef de la conception d'attribution de canal pour les réseaux maillés sans fil multi-radio se manifeste par le fait que le canal dépend fortement des liaisons logiques qui partagent un canal commun. Considérons le réseau montré dans la figure 3.5 où six canaux orthogonaux sont disponibles. Remarquez que toutes les liaisons (a, e), (e, d), (d, i) et (i, h) partagent le canal 3 et donc, si n'importe lequel des nœuds a, e, d, i, ou h décide de réassigner le canal sur ces liaisons virtuelles, alors le reste des liaisons doit changer leur attribution qui produit alors un ripple effect. Cette dépendance de canal entre les nœuds rend difficile le processus de réattribution.

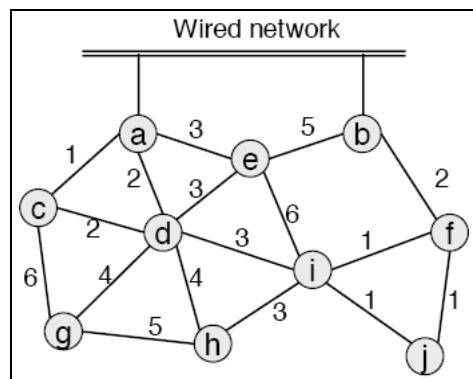
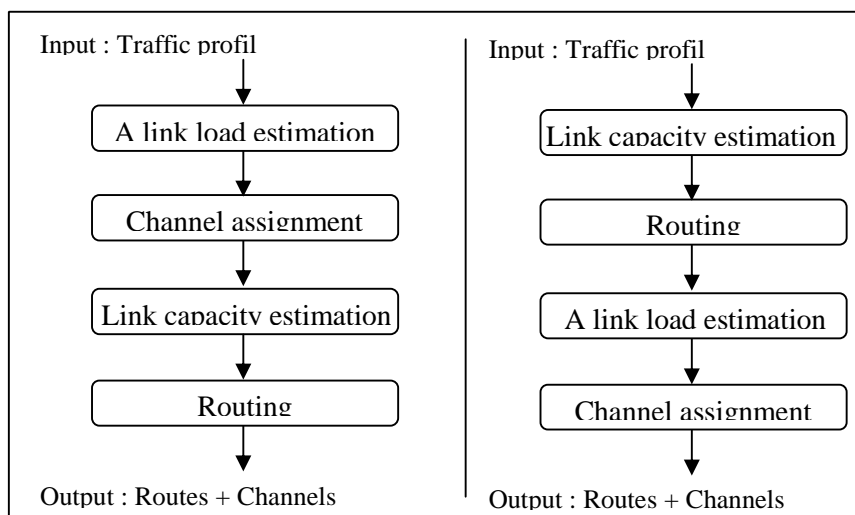


Figure 3.5 : Dépendance de canal

Finalement, un algorithme d'attribution de canal devrait prendre en considération la quantité de la charge de trafic sur les liaisons virtuelles. Il peut être assumé que chaque liaison virtuelle dans le réseau a la même charge de trafic. Cependant cela n'est pas vérifié pour la plupart des cas puisque certaines liaisons portent généralement plus de trafic que d'autres [25] (par exemple, les liaisons associées au nœud de passerelle). En général, on devrait donner plus de bande passante aux nœuds qui ont le plus haut trafic. Autrement dit, les canaux affectés à ces liaisons devraient être partagés entre un nombre limité de nœuds. Une telle stratégie d'attribution de canal consciente de trafic distribuerait les ressources de radio de telle sorte qu'elle corresponde à la distribution de la charge de trafic dans le backbone maillé sans fil.

Puisque l'attribution de canal dépend de la charge attendue de chaque liaison virtuelle, qui dépend à son tour du protocole de routage, une dépendance circulaire existe là entre l'attribution de canal et le routage [25]. Le routage dépend de la capacité de liaisons virtuelles, qui est décidée par la l'attribution de canal. C'est pour cela que la capacité d'une liaison virtuelle dépend du nombre des autres liaisons qui sont dans sa gamme d'interférence et qui utilise le même canal. En outre, l'attribution de canal dépend de la charge attendue des liaisons virtuelles, qui est affectée par le routage. Il y a deux stratégies différentes pour traiter cette circularité entre le routage et l'attribution de canal, comme montrer dans la figure 3.6.



**Figure 3.6 : Stratégies pour attribution de canal sensible à la charge de trafic**

Étant donné un ensemble de paires de nœud et la charge de trafic attendue entre chaque paire de nœud, selon la première stratégie montrée dans la figure 3.6 (a), l'algorithme de routage invente les itinéraires initiaux pour les paires de nœud. Donné ces itinéraires initiaux pour les paires de nœud et de la charge de trafic sur chaque liaison virtuelle, l'algorithme d'attribution

de canal assigne un canal à chaque interface radio prenant en compte la charge de trafic de la liaison. Cette attribution de canaux est finalement retransmise à l'algorithme de routage. La deuxième stratégie, montrée dans la figure 3.6 (b), diffère de la première dans le sens où l'algorithme de routage assume quelque attribution initiale de canaux aux interfaces radios. Basées sur cela, les capacités des liaisons sont évaluées et passées à l'algorithme de routage, qui passe à son tour la charge de liaison nécessaire pour l'attribution de canal. Évidemment, les deux stratégies peuvent terminer par des capacités de liaisons/charges de liaisons imprécises et alimentées à l'algorithme d'attribution de canal/ routage, qui peut exiger des itérations entre le routage et l'attribution de canal comme dans [25].

### 3.6 Attribution de canal et topologie du réseau

Dans un réseau maillé multi-canaux et multi-radio, l'attribution du canal aux différentes interfaces radio influence sur la topologie du réseau. Considérant un exemple d'un réseau avec 4 nœuds présenté dans la figure 3.7(a) et supposons que le nœud C est équipé par trois interfaces radio tandis que les autres nœuds (A, B et D) ont une seule interface radio. Chaque lien dans la figure est représenté par le numéro de canal utilisé. La figure 3.7(a) montre la topologie obtenue quand toutes les interfaces radio utilisent un même canal. La figure 3.7(b) montre le changement de la topologie, avec une autre attribution de canal.

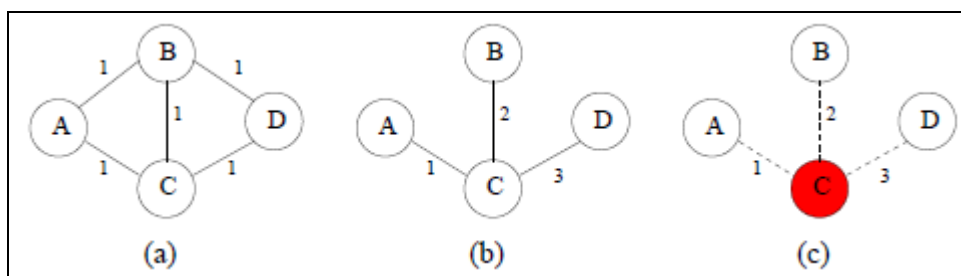


Figure 3.7 : Topologie du réseau avec variation d'attribution de canal

La modification de la topologie du réseau a trois inconvénients :

- Premièrement, la panne d'un sous-ensemble de noeuds, peut fortement causer un partitionnement du réseau. Par conséquent, des portions du réseau deviennent inaccessibles, ce qui mène a des pertes de flux comme montre la figure 3.7(c).
- Deuxièmement, les changements de topologie peuvent aboutir à des chemins sous optimaux entre des paires de noeuds en ce qui concerne certaines métriques, comme le débit, le retard, ou la fiabilité. Pour illustrer comment cela puisse arriver, considérez de nouveau la figure 3.7(a). Le noeud A communique avec le noeud B sur un chemin d'un seul saut. Après l'attribution de canal, Le noeud A communique avec B seulement sur un

chemin de deux sauts via C, comme indiqué dans la figure 3.7(b). Le choix d'un chemin avec plusieurs sauts n'est pas préféré pour trois raisons

- 1- Les chemins longs rapportent souvent des performances plus mauvaises que les chemins courts;
- 2- L'interférence d'un flux sur un chemin de nombre de saut élevé est naturellement plus grand; et
- 3- Un chemin plus long a une plus grande probabilité de défaillance.

Cependant, nous ne prétendons pas que tous les chemins longs vont probablement mal performer comparativement aux chemins courts parce que la performance de chaque alternative de chemin va probablement varier avec plusieurs facteurs comme le modèle de trafic, le placement de noeud, les caractéristiques des canaux et l'environnement. Néanmoins, nous soulignons qu'il est difficile de prévoir exactement, en pratique, l'attribution de canal et la topologie de réseau résultante qui ramèneront une performance optimale.

- Le troisième inconvénient de changer la topologie d'un réseau consiste en ce qu'il affecte les flux existants. Par exemple, supposons que le lien CD dans la figure 3.7 (b) est assigné un nouveau canal. Le processus d'attribution de canal doit être exactement coordonné; autrement, des cas peuvent surgir où une interface radio d'un bord d'un lien se commute à une nouvelle fréquence mais la deuxième interface ne fait pas parce qu'un message de contrôle est soit perdu ou retardé. Par conséquent, n'importe quels flux de D au reste du réseau qui a existé au moment de l'attribution de canal seront perturbés pendant la commutation. Il est difficile de surmonter de tels cas en pratique parce que la configuration des interfaces radios exige la coordination synchronisée de temps entre les routeurs pendant l'attribution de canal.

### **3.7 Attribution du canal, connectivité et interférence**

Un noeud routeur doit partager un canal commun avec chacun de ses voisins dans la gamme de communication, exigeant ainsi la fondation d'une liaison virtuelle. De plus, pour réduire l'interférence de réseau, un noeud devrait réduire au minimum le nombre de voisins avec qui il partage un canal commun. Il existe donc un compromis entre la maximisation de la connectivité et la minimisation de l'interférence. Ce compromis est illustré par l'exemple dans la figure 3.8. La figure 3.8 (a) montre La connectivité du réseau quand un seul canal fonctionne sur une seule interface radio. Dans ce scénario, une liaison est placée entre deux

noeuds s'ils sont dans leurs gammes de transmission respectives. C'est la connectivité réalisable maximale de réseau puisqu'un seul canal commun est partagé entre tous les noeuds. Maintenant, concentrons-nous sur le scénario de multi-radio multi-canal représenté dans les Figures 3.8 (b) et (c). Il y a quatre canaux disponibles pour la communication, étant donné que chaque noeud est équipé avec deux interfaces radios. Nous illustrons un cas où la connectivité de réseau est maximisée (comme celle dans le cas d'une seule interface radio et d'un seul canal) et un autre cas où l'interférence est réduite au minimum (avec l'utilisation efficace des canaux disponibles) et nous expliquerons aussi comment l'un affecte l'autre. Dans la figure 3.8 (b), l'attribution des canaux aux interfaces radios aboutit à une connectivité maximale de réseau. Cependant, cela ne peut pas être réalisé à moins que trois des quatre canaux disponibles ne soient assignés et trois des liaisons soient assignées au même canal (c'est-à-dire le canal 2). Par exemple, il y a une liaison de communication directe entre chaque paire de voisins. Cependant, toutes les liaisons ne peuvent pas être actives simultanément à cause de l'interférence possible. Parallèlement, la figure 3.8 (c) montre comment l'interférence pourrait être complètement éliminée et toutes les liaisons peuvent être simultanément actives. Le compromis ici est qu'il n'y a aucun canal commun entre les voisins, b et d.

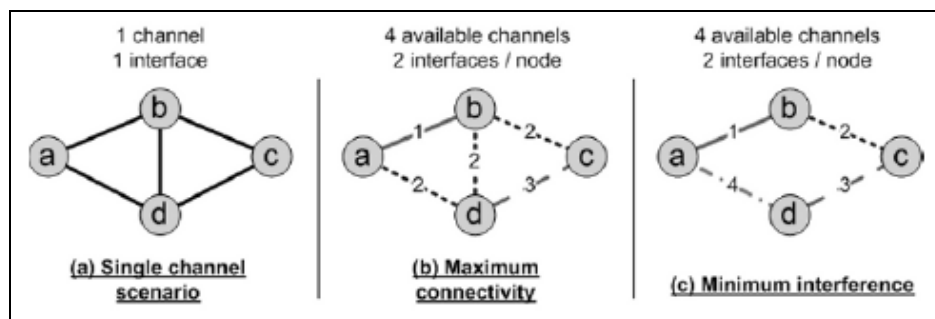


Figure 3.8 : Compromis entre la connectivité et l'interférence

L'exemple précédent illustre clairement que le but de l'attribution de canal est de réaliser un équilibre entre (i) minimisation de l'interférence (sur n'importe quel canal donné) et (ii) maximisation de la connectivité. Dans ce sens, l'attribution de canal dans un réseau sans fil multi-sauts peut être vue comme un problème de contrôle de topologie [35] (semblable au contrôle de puissance de transmission, par exemple). À la différence d'un réseau câblé, les liaisons dans un réseau sans fil sont flexibles et peuvent être accordées ou configurées. Les paramètres tenables dans un environnement sans fil incluent la fréquence de canal, la puissance de transmission, le taux de bit et la transmission directionnelle (utilisant des antennes directionnelles). En général, le contrôle de topologie exploite ces paramètres pour obtenir une topologie désirable du réseau. Cela peut être un des rôles d'attribution de canal

dans les réseaux maillés sans fil en plus de la maximisation de la connectivité et la minimisation de l'interférence.

### 3.8 Classification d'attribution de canal

Selon la fréquence d'attribution de canal, les auteurs [22] [23] [24] classifie les stratégies d'attribution de canal en trois groupes : statique, dynamique et hybride.

#### 3.8.1 Statique

La stratégie d'attribution statique assigne chaque canal à une interface radio de manière permanente. L'attribution statique peut être de nouveau classifiée en deux types :

##### *1) Approche de canal commun*

Dans cette approche, les interfaces radio de tous les noeuds sont assignées à un ensemble commun de canaux (par exemple si deux interfaces sont utilisées dans chaque noeud, donc les deux interfaces sont affectées aux deux mêmes canaux à chaque nœud). L'avantage de cette approche est que la connectivité du réseau est la même comme celui de l'approche d'un seul canal. Notez que le scénario où un seul canal et une seule interface est un cas spécial de la stratégie d'attribution de canal statique commune.

##### *2) Approche de changement de canal*

Dans cette approche, les interfaces des noeuds différents peuvent être affectées à un ensemble différent de canaux (par exemple [25]). Avec cette approche, il y a une possibilité que la longueur des chemins entre les noeuds puisse augmenter. En plus, si l'attribution des canaux aux interfaces ne soit pas faite soigneusement, des partitions de réseau peuvent surgir. D'après [24] et [29], les stratégies d'attribution de canal statique avec changement de canal peuvent être classées en deux catégories : solutions basées sur la topologie [17] et décision conjointe entre routage et attribution de canal [25] [26].

- Les solutions à base de topologie définissent une certaine métrique topologique pour mesurer l'interférence, comme le nombre de voisins utilisant le même canal et réduire ensuite au minimum cette métrique. Cependant, c'est plutôt le trafic que la topologie qui cause l'interférence. Ces solutions ne fonctionnent donc pas bien si le trafic de réseau n'est pas équilibré [25] [26].
- Les solutions de co-décision (basé trafic), par exemple, l'algorithme Hyacinth proposé dans [25], assume que la demande de trafic de bout en bout est stable à long terme et détermine conjointement l'affectation de canal et le routage dans le réseau en itérant ces

deux étapes plusieurs fois, jusqu'à ce que la demande de trafic et la capacité des liens soient bien adaptées.

Les stratégies d'attribution de canal statiques sont bien convenues quand le temps de commutation d'interface est grand. Avec l'attribution statique, les noeuds qui partagent un canal sur une de leurs interfaces peuvent directement communiquer, tandis que d'autres ne peuvent pas. Ainsi, l'effet d'attribution de canal statique est de contrôler la topologie de réseau en décidant quels sont les noeuds qui peuvent communiquer entre eux.

### 3.8.2 Dynamique

La stratégie d'attribution dynamique permet à n'importe quelle interface d'être assignée à n'importe quel canal et les interfaces peuvent fréquemment commuter d'un canal à un autre. Dans cette stratégie, deux noeuds qui doivent communiquer ont besoin d'un mécanisme de coordination pour assurer qu'ils sont sur un canal commun à un certain temps. Par exemple, le mécanisme de coordination peut exiger que tous les noeuds visitent un canal "de rendez-vous" commun périodiquement (par exemple [27]). L'avantage de l'attribution dynamique est la capacité de commuter une interface à n'importe quel canal, offrant ainsi la possibilité de couvrir beaucoup de canaux avec peu d'interfaces. Le défi major avec cette stratégie de commutation dynamique est de coordonner les décisions de quand commuter les interfaces ? Aussi de savoir quel canal commuter les interfaces pour tous les noeuds dans le réseau ?

### 3.8.3 Hybride

La stratégie d'attribution hybride combine les stratégies d'attribution statique et dynamique en appliquant une attribution statique pour quelques interfaces et une attribution dynamique pour d'autres interfaces [28] [32]. Les stratégies hybrides peuvent être divisées en deux ensembles, basés sur la nature des interfaces : les interfaces fixes utilisant l'approche de canal commun [28] ou les interfaces utilisant l'approche de variation de canal [23] [32]. Les interfaces fixes peuvent être assignées à un canal de contrôle consacré [28] ou à un canal de contrôle et de données [32], tandis que les autres interfaces peuvent être commutées dynamiquement sur les autres canaux.

Les stratégies d'attribution hybrides sont attirantes parce que, comme avec l'attribution fixe, elles utilisent des algorithmes de coordination simples, en conservant toujours la flexibilité de l'attribution de canal dynamique.

La figure suivante illustre les différentes classes de stratégie d'attribution de canal :

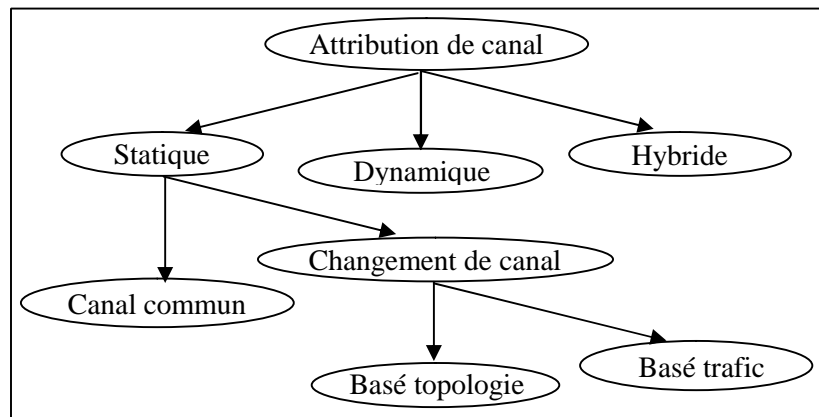


Figure 3.9 : Classification des stratégies d'attribution de canal

### 3.8.4 Résumé des stratégies d'attribution de canal

Comme les noeuds dans un réseau sans fil multi-canaux multi-radio communiquent entre eux sur plusieurs canaux sans fil dans un mode multi-sauts, cela nécessite une attribution de canal appropriée et efficace est un problème de recherche très stimulant et critique.

Plusieurs stratégies d'attribution de canal ont été proposées, le tableau suivant résume les avantages et les inconvénients de chaque stratégie.

Tableau 3.1 : Comparaison des stratégies d'attribution du canal

Stratégie		Avantages	Inconvénients	
S t a t i q u e	Canal commun	- Meilleure connectivité - Gain de temps	- Ne bénéficie pas de tous les canaux disponibles	
	Change ment de canal	Basé topologie	- Gain de temps - Interférence minimale	- Si la charge de trafic augmente, le taux d'interférence augmente.
		Basé trafic	- Gain de temps - Interférence minimale - Proportionnel à la charge de trafic	- Applicable que si la charge de trafic est stable, sinon devient dynamique
Dynamique		- Bénéficie de tous les canaux disponibles avec peu d'interface.	- Temps de commutation - Possibilité de problème de surdité (réseau partitionné), Besoin d'un mécanisme de coordination - Diffusion coûteuse.	
Hybride		- Bénéficie de tous les canaux disponibles avec peu d'interface. - Assure la connectivité	Sous utilisation des ressources réseau.	

### 3.9 Travaux liés

#### 3.9.1 MR-LQSR

Le protocole Multi-Radio Link Quality Source Routing (MR-LQSR) [5] a été développée par Microsoft pour des réseaux communautaires sans fil statiques. Le protocole travaille en accord avec la Couche de Connectivité de Maille (MCL). Le MCL permet aux applications de la couche supérieure de se connecter au réseau maillé sans fil en utilisant le WiFi ou WiMAX. Le MCL met en oeuvre une couche d'interposition entre les couches réseau et liaison. Le routage des paquets est effectué par MR-LQSR, qui est une version optimisée du protocole Dynamic Source Routing (DSR). Le protocole MR-LQSR suppose que le nombre d'interfaces sans fil est égal au nombre de canaux étant utilisées dans le réseau. Le protocole identifie tous les nœuds dans le réseau maillé sans fil et assigne des poids à toutes les liaisons possibles. Pour faire ainsi, les informations de liaison incluant l'attribution de canal, la bande passante et les taux de perte sont propagés à tous les nœuds dans le réseau. Cette propagation est combinée avec la livraison de paquets de contrôle de DSR. Le Temps de Transmission Attendu (ETT) sur chaque liaison est calculé en utilisant le nombre de Transmission Attendu (ETX), la bande passante et la perte de paquets. La métrique ETT est utilisé pour calculer la métrique WCETT (weighted cumulative expected transmission time), qui définit la métrique du chemin conçu pour WMNs multi-radio. Le WCETT est alors appliqué au plan de cache de liaison du protocole DSR. Dans DSR d'origine, le coût par défaut de liaisons est mis à un. De là, quand l'algorithme Dijkstra est exécuté sur la cache de liaison par un nœud source, le chemin le plus court en termes de nombre de sauts est toujours rendu. Cependant, quand le WCETT est utilisé comme coût de liaison, le protocole vise à rendre le chemin en termes de bande passante de liaison, taux de perte et diversité de canal.

#### 3.9.2 Hyacinth

Hyacinth [26] est un protocole multi-canaux pour les réseaux maillés sans fil statique. Il utilise des interfaces radios et des canaux multiples pour améliorer la performance du réseau. Il utilise un algorithme d'attribution de canal entièrement distribué, qui peut s'adapter dynamiquement avec la charge de trafic.

Puisque la majorité de trafic dans les réseaux maillés sans fil est orienté vers/depuis Internet. Les auteurs [26] et [30] utilisent un algorithme de routage fondé sur un arbre recouvrant pour équilibrer la charge dans le réseau. Les nœuds routeurs ayant accès aux réseaux câblés (les

passerelles) sont considérés comme les nœuds racines des arbres. La métrique de construction de cet arbre est fonction de la charge de trafic d'un nœud et sa distance de la passerelle.

L'algorithme d'attribution de canal fonctionne en deux phases:

- Attribution des interfaces aux voisins

Dans cette phase, chaque nœud sépare ses interfaces en deux ensembles: UP-NIC et DOWN-NIC. Les UP-NIC sert à lier le nœud avec son nœud parent, quand aux DOWN-NIC le relie avec ses nœuds enfants. Chaque nœud est responsable pour changer les canaux de ses DOWN-NIC seulement. Raniwala juge que cette phase permet de résoudre le problème de dépendance des canaux en cas de réattribution.

- Attribution des canaux aux interfaces

Au cours de cette phase, chaque nœud échange périodiquement un message HELLO, contenant l'état d'utilisation des canaux avec ses voisins dans la gamme d'interférence. Cette information donne la charge totale d'un canal, qu'un nœud peut l'utiliser afin de basculer vers le canal le moins utilisé.

Etant donné que le modèle de trafic et la charge des canaux change avec le temps, la deuxième phase doit être ajusté périodiquement pour équilibrer la charge entre les nœuds et assigne les canaux de tel sorte que la bande passante disponible pour chaque lien soit proportionnel à sa charge de trafic.

Dans [31], l'auteur donne une solution plus performante que celle de hyacinth au problème de dépendance des canaux. Elle vise à classer les canaux en plusieurs groupes. Un changement de canal par la suite affecte les nœuds qui utilisent les canaux qui appartiennent au même groupe seulement.

### 3.9.3 MCR

Le protocole Multi Channel Routing (MCR) [32] a été développé pour les WMNs dynamique, où les nœuds ont des interfaces radio multiples, chaque'un d'eux peut travailler avec des canaux multiples. Le protocole utilise un mécanisme de commutation d'interface pour assigner les interfaces aux canaux. Deux types d'interfaces sont assumés : fixe et commutable. Dans les interfaces fixes, il y a K interfaces parmi M interfaces en total sont assumés pour fonctionner sur K canaux fixe. Dans les interfaces commutables, les interfaces restantes sont dynamiquement assignées à n'importe lequel des canaux restants. La commutation est effectuée selon le nombre maximal de paquets de données mis en file d'attente pour un canal. Des files d'attente multiples sont maintenues pour toutes les interfaces commutables. Chaque nœud maintient une table de voisinage et une liste d'utilisation du canal.

- La table de voisinage contient les informations relatives aux canaux fixes utilisées par les nœuds voisins.
- La liste d'utilisation de canal contient le nombre des nœuds qui utilisent chaque canal comme leur canal fixe dans le voisinage de deux sauts.

Chaque nœud transmet périodiquement un paquet HELLO sur tous les canaux, contenant le nombre de canaux fixes du nœud. Chaque nœud recevant ce paquet HELLO met à jour sa table de voisinage et la liste d'utilisation de canal. Les informations de la table et de la liste sont utilisées pour contrôler le mécanisme de commutation de canal et d'interface. Le mécanisme de commutation aide le MCR dans la découverte de chemins sur plusieurs canaux. MCR utilise une nouvelle métrique de routage, qui est calculée en fonction de diversité de canal, de coût de commutation d'interface et de nombre de sauts.

- Le coût de diversité est assigné selon le nombre minimal de canaux utilisés dans un itinéraire. Ainsi, on considère un itinéraire avec un plus grand nombre de canaux distincts dans un itinéraire d'avoir un coût de diversité inférieur.
- Le coût de commutation est utilisé pour réduire au minimum la commutation fréquente des interfaces radio.

Le mécanisme de découverte d'itinéraire de MCR est semblable à celui de DSR. De plus, chaque RREQ contient aussi le nombre de canal et le coût de commutation. Quand la destination reçoit le RREQ, elle calcule le coût de diversité (le nombre de canaux dans le RREQ) et le coût de commutation (la somme des coûts de commutation de toutes les liaisons). Ces coûts aident la destination dans la détermination et la sélection du chemin optimal disponible entre la source et la destination.

### 3.9.4 AODV-ST

Comme hyacinth, AODV-ST [34] utilise un algorithme de routage (IACA : Interference Aware Channel Assignment) [28] basé sur un arbre. Mais son but d'attribution de canal est de réduire l'interférence externe entre les nœuds routeurs et les autres réseaux. IACA fixe une interface dans chaque nœud et lui attribue un canal fixe. Ce canal assure :

- La connectivité des nœuds du réseau,
- la diffusion, et
- Dans le cas de défaillance d'un lien, sert de lien de redirection.

Cet algorithme est basé sur une extension du concept de graphe de conflit, appelé le Graphe de Conflit Multi-radio (MCG). Cet algorithme estime l'interférence et la bande passante, en se

basant sur le nombre d'interfaces radio interférente sur chaque canal soutenu par chaque routeur, où une interface radio interférente est une interface radio simultanément opérationnel et qui est visible à un nœud routeur, mais externe à son réseau. L'algorithme d'attribution de canal travaille sur une stratégie à base de rang où le rang pour chaque canal disponible est basé sur l'interférence et la charge de canal. On considère la charge, cependant, pour les réseaux sans fil externes seulement.

### 3.9.5 MesTic

Le but principal d'une attribution de canal est l'amélioration de la bande passante du réseau. MesTic '*Mesh based Traffic and interfere aware Channel assignment scheme*' [33] est un algorithme d'attribution de canal centralisé, basé sur une fonction de classement qui prend en considération : la charge de trafic, le nombre de saut à partir de la passerelle et le nombre d'interfaces dans un nœud.

Le classement d'un nœud détermine sa priorité dans l'attribution de canal, il est dynamique car il est calculé en fonction de :

- La charge de trafic d'un nœud obtenu à partir du modèle de trafic du réseau maillé qui est calculé en fonction de nombre de flux qui passe par ce nœud comme dans [25].
- La distance d'un nœud, mesuré en terme de nombre de sauts minimum entre le nœud et la passerelle.
- Nombre d'interfaces radio disponible pour un nœud.

$$\text{Rang} = \frac{\text{Charge du trafic}}{\text{Min-Nb-Saut (passerelle, nœud)} * \text{Nb-interface-noeud}} \quad (1)$$

Il est clair que la charge du trafic passant à travers un noeud a un impact sur l'attribution de canal. Si un noeud relais plus de trafic, en lui attribuant le canal de moins interférence pour augmenter le débit du réseau. Ainsi, le trafic global dans le numérateur dans l'Equation (1) augmente le rang d'un noeud avec plus de trafic. Aussi, en raison de la nature hiérarchique de la topologie mesh, les nœuds qui sont plus proches de la passerelle doivent avoir la préférence dans l'attribution de canal, ils sont plus susceptibles d'exercer plus de trafic. En même temps, le nombre de radios sur un noeud donne une flexibilité dans l'affectation de canal et affecte inversement sa priorité (c'est-à-dire, plus petit est le nombre d'interfaces radios, plus élevée sera la priorité dans l'attribution de canal).

Une fois le rang de chaque nœud est calculé, l'algorithme parcourt les nœuds du réseau par ordre décroissant de leurs rangs et affecte les canaux à leurs interfaces radio.

MesTic réaffecte les canaux dans deux cas : soit le changement dans les caractéristiques de la charge de trafic ou bien le changement de topologie du réseau par l'ajout ou le départ des nœuds dans le réseau.

### 3.9.6 Attribution de canal avec QoS

Nous avons trouvé un seul article traitant le problème d'attribution de canal avec qualité de service [17], basé topologie 'INSTC' L'auteur dans [17], propose un algorithme nommé INSTC 'minimum INterference Survivable Topology Control' qui fait une attribution statique et vise à trouver une topologie de réseau avec taux d'interférence minimal. A partir de la topologie physique d'un réseau, l'algorithme calcule pour chaque lien l'ensemble des liens qui s'interfère potentiellement avec lui.

On dit que le lien  $e = (u, v)$  s'interfère potentiellement avec le lien  $e' = (x, y)$  si et seulement si

$$x \in Du \cup Dv \text{ ou bien } y \in Du \cup Dv \text{ de même } u \in Dx \cup Dy \text{ ou bien } v \in Dx \cup Dy$$

Tel que  $Dx$  : représente le disque d'interférence du nœud  $x$  (disque centré a  $x$  avec un rayon  $R > r$  où  $r$  est le rang de transmission).  $Dx$  dénote l'ensemble des nœuds couvert par ce disque. Pour faire l'attribution de canal, l'algorithme classe tous les ensembles d'interférence potentiel par ordre décroissant, et commence à former un sous graphe contenant tous les liens du plus petit ensemble d'interférence potentiel, puis il affecte les canaux aux interfaces radio de leurs nœuds de telle manière à réduire l'interférence.

Après avoir terminé cet algorithme, le résultat trouvé est une topologie de réseau avec un minimum d'interférences. Cette attribution sera maintenue jusqu'à l'ajout ou la panne d'un nœud du réseau.

L'article présente par la suite un algorithme optimal (BAR : Bandwidth Aware Routing) pour résoudre le problème de routage avec QoS, qui cherche à allouer de la bande passante pour des demandes de connexion sans altérer les connexions en cours.

### 3.9.7 Synthèse des protocoles MC-MR

Les caractéristiques les plus importantes des protocoles d'attribution de canal existant pour les réseaux maillés sans fil sont récapitulées dans le tableau 3.2.

On remarque que les critères de base pour l'attribution de canal sont bien la connectivité, le contrôle de topologie, la minimisation d'interférence, la charge de trafic et la destination finale des flux.

Tableau 3.2 : Etude comparative des protocoles d'attribution de canal existants

	Classe d'attribution de canal	Critère d'attribution	Métrique	Temps de commutation	Connectivité	topologie	Dist. finale
<b>MR-LQSR</b>	Statique avec canal commun	Attribution identique	WCETT	Pas de commutation	Meilleur connectivité	Fixe	N'importe quel noeud
<b>Hyacinth</b>	Dynamique	Charge de trafic	Nombre de saut, charge de chemin/lien	Commutation non fréquente	Assurer par l'attribution de canal	Définie par arbre de routage	La passerelle
<b>AODV-ST</b>	Hybride	Interférence interne et externe	ETT	Commutation non fréquente	Assurer par un radio par défaut	Définie par arbre de routage	La passerelle
<b>MesTic</b>	Statique avec variation de canal	Charge de trafic et interférence	Nombre de saut, charge de chemin/lien	Pas de commutation	Assurer par un radio par défaut	Fixe	N'importe quel noeud
<b>MCP</b>	Hybride	charge de trafic	Diversité de canaux, coût de commutation	Considérer par la métrique	Assurer par commutation de canal	Change dynamiquement	N'importe quel noeud
<b>INSTC</b>	Statique avec variation de canal	Interférence		Pas de commutation	Assurer par l'attribution de canal	Définie par l'attribution de canal	N'importe quel noeud

### 3.5 Conclusion

Dans ce chapitre, nous avons identifié les défis clefs associés à l'attribution des canaux aux interfaces radio dans les réseaux maillés multi-radio et multi-canaux. Après avoir présenté une taxonomie des stratégies d'attribution de canal existantes, nous avons étudié des exemples de protocole pour chaque stratégie.

Dans le chapitre qui suit, nous allons présenter notre nouveau protocole pour l'attribution du canal et le routage qui est sensible aux types de trafic dans le réseau.

PARTIE II  
CONTRIBUTION

---

## Attribution de canal avec différentiation de service

---

### 4.1 Introduction

La nouvelle génération de réseaux sera construite par l'intégration de réseaux filaires et sans fil. Les réseaux sans fil ad hoc (MANET), les réseaux Sensor (WSN), les réseaux cellulaires et les réseaux fixes existants seront intégrés ensemble pour fournir ce qu'on appelle un réseau maillé sans fil (WMN). Cet environnement fournira à l'utilisateur la capacité de communiquer à tout moment et n'importe où. Dans un tel environnement, de nouvelles applications seront créées. En même temps, le nombre des utilisateurs qui gagnent l'accès au réseau augmentera aussi. Une exigence importante de ces applications est la demande de garanties de la qualité de service.

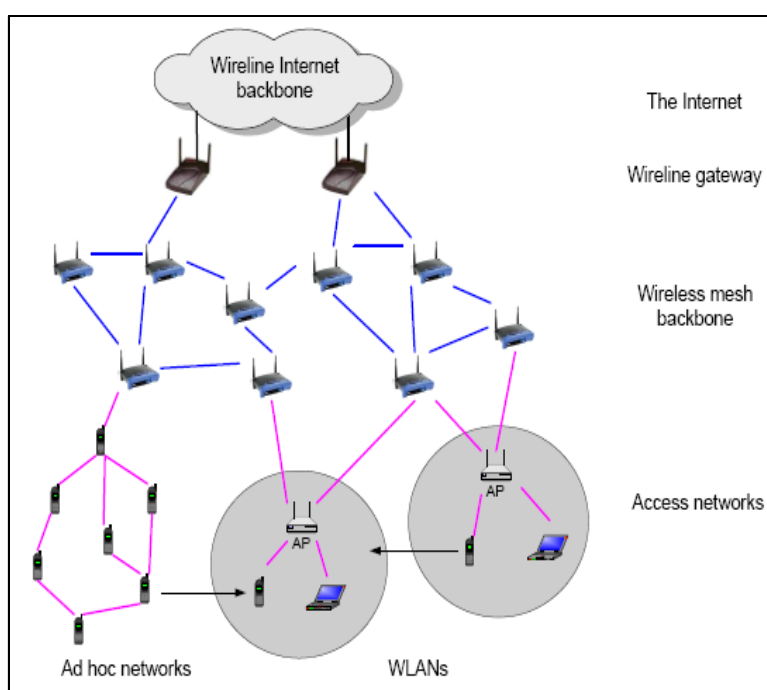


Figure 4.1 : Architecture du réseau à trois niveaux

La figure 4.1 illustre l'architecture du réseau de future génération pour l'accès sans fil à haut débit, qui consiste en des passerelles filaires, des routeurs sans fil, et des stations mobiles (MSs), organisé dans une architecture à trois niveaux.

La passerelle filaire est connectée au backbone Internet. Les routeurs sans fil sont des sites fixes et forment le backbone sans fil. Les MSs accèdent à Internet via des routeurs sans fil d'une manière distribuée (dans les réseaux ad hoc) ou centralisée (par le biais des points d'accès (AP) dans les réseaux locaux sans fil.

Dans le backbone sans fil, le canal devient un goulot d'étranglement en raison de la haute utilisation d'un tel canal. En outre, les nœuds ne peuvent pas recevoir et transmettre des données en même temps en utilisant un seul canal. Pour surmonter ces limitations, des canaux multiples et des interfaces réseaux multiples peuvent être équipés pour chaque routeur. Cependant, plusieurs interfaces dans chaque nœud exigent une planification d'attribution de canal. Les canaux doivent être assignés de telle sorte que l'interférence soit diminuée et les performances du réseau soient augmentées en même temps pour servir les exigences des différentes applications. Puisque le nombre des canaux disponibles est limité, il est désiré qu'une attribution et une réattribution des canaux soient faites dynamiquement pour le support de qualité de service.

Dans ce chapitre, nous proposons un nouveau mécanisme d'attribution de canal avec différenciation de service et un protocole de routage basé sur de plusieurs arbres recouvrants, basé sur le type de trafic et améliore la qualité de service et l'efficacité de l'équilibrage de charge ainsi que la tolérance aux pannes pour l'architecture du backbone sans fil [46].

## 4.2 Motivation

Avec la croissance et l'omniprésence de l'Internet constatée ces dernières années, de nouvelles applications sont envisagées. Celles-ci présentent de nouveaux types de trafic et de nouvelles exigences qui demandent à leur tour de nouveaux services du réseau qui conviennent à ces caractéristiques et exigences. Tant le comité IEEE 802 que l'IETF ont proposé des protocoles et des mécanismes qui permettraient à Internet de fournir les services nécessaires pour les nouvelles applications [39].

L'IETF a proposé deux grandes approches pour assurer la qualité de service dans Internet: des services intégrés (IntServ) et des services différenciés. Bien que IntServ peut fournir des garanties de qualité de service à grains fins, l'information de réservation par flux et la lourde charge augment les frais de signalisation. Il est à constater qu'IntServ ne convient pas pour des réseaux à grande échelle. En outre, au moyen d'un gros modèle de différenciation,

l'approche DiffServ est une solution évolutive avec des frais de signalisation réduits. Dans les routeurs de bord de type DiffServ, les paquets peuvent être classés en un nombre limité de classes de service suivant le Service Level Agreement (SLA) qui est négocié avec le Fournisseur de service Internet (ISF). Dans un routeur de cœur, les paquets provenant de différentes classes sont globalement différenciés par différents comportements par-sauts (PHBs : per-hop behaviors). Le bord du réseau prend les fonctionnalités complexes telles que la classification de trafic et le conditionnement, et le réseau de base est conservé simple, ce qui rend DiffServ évolutive.

La recherche sur l'approche DiffServ a principalement porté sur Internet filaire. Dans ce chapitre, nous appliquons la DiffServ au backbone sans fil, qui est constituée d'un ensemble de routeurs équipés par plusieurs interfaces radio et opèrent sur plusieurs canaux.

Une plate-forme de DiffServ est une approche prometteuse pour connecter des réseaux hétérogènes sans fil/filaire avec le backbone d'Internet pour fournir une QoS de bout en bout. Dans la DiffServ sans fil, chaque routeur agit comme un routeur de bord pour l'APs ou les utilisateurs mobiles sous sa couverture. Le routeur sans fil rassemble les exigences de service des utilisateurs sous sa couverture et les agrège à une exigence SLA au backbone sans fil. Un routeur sans fil travaille aussi comme un routeur principal (de cœur), c'est-à-dire qu'il retransmet le trafic de\aux passerelles filaires. Tous les routeurs sans fil utilisent plusieurs files d'attente séparées contrôlées par un certain algorithme d'ordonnancement pour fournir des classes de service différencié [15]. Les passerelles filaires sont les routeurs de passerelle fournissant l'interface avec la DiffServ de backbone Internet.

Le protocole de routage avec QoS dans un routeur sans fil cherche les ressources disponibles pour chaque classe soutenue du trafic global (classe EF, AF ou BE) en raison de l'exigence de QoS différente d'EF et AF. Le routeur sans fil représente les flux de trafic qui viennent de ses réseaux connectés et ont des exigences de QoS semblables par classe (EF ou AF), les regroupe et les achemine ensemble dans le même chemin. Le backbone mesh sans fil peut contenir plusieurs passerelles au backbone d'Internet. La destination de trafic venant de n'importe quel routeur sans fil doit être l'une de ces passerelles.

Dans notre cas, nous travaillons avec des nœuds équipés chacun par plusieurs interfaces réseaux. Chaque interface réseau est associée avec une seule file d'attente. Un nœud routeur après qu'il classe le trafic, il envoie chaque trafic vers l'interface réseau appropriée. A la différence de la DiffServ originale qui gère plusieurs files d'attente pour chaque nœud par des algorithmes d'ordonnancement, notre protocole gère les interfaces réseaux de chaque nœud par un algorithme d'attribution de canal.

### **4.3 Architecture du système et formulation de problème**

#### **4.3.1 Architecture du système**

Un réseau maillé sans fil consiste en un ensemble de nœuds dont chacun est équipé par plusieurs interfaces radio et joue les rôles de points d'accès au routeur intermédiaire ou aux deux à la fois. Typiquement un WMN est connecté à l'Internet filaire à travers une ou plusieurs passerelles et la plupart des paquets traversant le réseau passent par une de ces passerelles.

Puisque les nœuds du WMN sont plus stationnaires, le protocole de routage utilisé dans WMNs est plus semblable à ceux utilisés dans les réseaux filaires que ceux utilisés dans les réseaux mobiles ad hoc (MANET). Plus précisément, parce que le nombre maximum de sauts dans WMNs est limité à un petit nombre (environ 5) [38], de nombreux protocoles de routage WMN empruntent en réalité la conception de protocoles de routage utilisés dans les réseaux filaires locaux, qui sont basés sur le concept d'arbre, comme c'est le cas du protocole IEEE 802.1D [40]. La solution décrite dans ce chapitre s'inspire de cette idée. Plus particulièrement, elle utilise différents arbres de routage (un arbre pour chaque classe de trafic) comme dans [41].

Le backbone sans fil peut être groupé en plusieurs domaines locaux [30] [10], selon la relation entre les routeurs mesh et les passerelles. Chaque domaine est dirigé par une passerelle. Les chemins relient les différents routeurs avec la passerelle et forment une topologie en arbre. Dans notre étude, on s'intéresse à un seul groupe. Le réseau sera donc formé par un ensemble de routeurs et une seule passerelle.

#### **4.3.2 Objectif de l'attribution de canal**

Les futurs réseaux haut débit porteront beaucoup de flux concurrents avec des exigences diverses. De là, il est crucial que la bande passante du réseau et les autres ressources du réseau soient partagées efficacement et équitablement entre tous les flux concurrents. Dans notre étude, nous allons considérer que deux classes de flux : QoS-flux (ou bien EF : Expected Forwarding) qui portent des exigences de QoS de bout en bout sur le chemin établi et des flux de Best effort (BE) qui n'ont pas d'exigences de QoS spécifiques sur le chemin établi.

La métrique de routage appropriée pour chaque classe de trafic est différente. Par exemple, le trafic de la classe QoS de la plus haute priorité demande un délai minimum et une bande passante maximale. Le trafic de la classe BE devrait éviter la famine autant que possible.

Notre but dans l'attribution de canal est de maximiser l'utilisation de toutes les ressources du réseau pour servir le maximum de demandes avec QoS tout en évitant le problème de famine des demandes de la classe BE.

### 4.3.3 Formulation de problème

Nous considérons un réseau maillé fixe avec plusieurs nœuds et une passerelle, à chaque nœud, il est attribué un identifiant unique ID et équipé par plusieurs interfaces radio, dont chaque une s'opère sur un canal différent. Il y a  $CH$  canaux orthogonaux dans le réseau, dénotés par l'ensemble  $CH = 1, 2, \dots, C$ . Dans le standard IEEE 802.11a,  $C = 12$ . Le nombre de canaux dans le réseau est supérieur au nombre d'interfaces radio dans chaque nœud  $v$ ,  $1 \leq R(v) \leq C$ . La portée de communication est identique pour tous les nœuds, ce qui rend la relation de voisinage symétrique.

Le réseau est représenté par un graphe non orienté  $G(V, E)$  où  $V$  représente l'ensemble des nœuds dans le réseau,  $E$  est l'ensemble des liaisons physiques entre les nœuds, une liaison  $e(u, v) \in E$  implique que  $u$  est dans la gamme de communication de  $v$ . Dans ce cas, les nœuds  $u$  et  $v$  sont appelés des nœuds voisins directs (à un saut).

Les notations utilisées dans cette section sont listées dans le tableau suivant.

**Tableau 4.1 : Table des notations**

Notation	Description
$V$	L'ensemble des nœuds du réseau.
$N$	Nombre de nœuds du réseau. $N =  V $
$E$	L'ensemble des liens physiques.
$G$	Graphe du réseau.
$t(e), e \in E$	L'ensemble des nœuds dans le voisinage de transmetteur de $e$ .
$r(e), e \in E$	L'ensemble des nœuds dans le voisinage de receveur de $e$ .
$CH$	Nombre de canal dans le réseau.
$ID(v), v \in V$	Identificateur unique du nœud $v$ .
$R(v), v \in V$	Nombre d'interfaces radios d'un nœud $v$ .
$N(v), v \in V$	L'ensemble des voisins directs d'un nœud $v$ .
$CL(v), v \in V$	L'ensemble des nœuds enfants d'un nœud $v$ .
$P_{s,d}, s, d \in V$	Le chemin entre le nœud $s$ et le nœud $d$ .
$f(e_{u,v}), e_{u,v} \in E, u, v \in V$	Le flux passant par la liaison $e$ entre le nœud $u$ et le nœud $v$ .
$W_e^k, k \in CH, e \in V$	La bande passante du canal $k$ sur la liaison $e$

Supposons  $s_t^k(e_{uv})$ , une variable binaire qui dénote l'état actif d'une liaison sur le canal  $k$  ( $k \in CH$ ) sur l'arc reliant le nœud  $u$  au nœud  $v$  pendant un slot de temps  $t$ .

$$s_t^k(e_{uv}) = \begin{cases} 1 & \text{Si le lien } e_{uv} \text{ est active sur le canal } k \text{ dans le slot de temps } t \\ 0 & \text{Sinon} \end{cases}$$

Une attribution de canal optimale doit satisfaire les trois contraintes suivantes (contrainte de liaison, contrainte d'interface radio d'un nœud et contrainte d'interférence) [30]:

#### 4.3.3.1 Contrainte de liaison

Le nombre total de liaisons actives sur un arc pendant un slot de temps  $t$  ne peut pas être plus grand que  $h(e_{uv})$ , où  $h(e_{uv})$  est le nombre maximal de liaisons permises sur un arc pendant un slot de temps. La contrainte est donnée par l'équation (1) :

$$\sum_{k \in CH} s_t^k(e_{uv}) \leq h(e_{uv}), \forall e_{uv} \in E \quad (1)$$

#### 4.3.3.2 Contrainte d'interface radio d'un nœud

Le nombre total des liaisons actives d'un nœud sur tous les arcs adjacents ne peut pas être plus grand que le nombre maximal d'interfaces radio de chaque nœud. Supposons que  $N(u)$  dénotent l'ensemble des voisins du nœud  $u$ . Alors la contrainte est exprimée comme indiqué ci-dessous :

$$\sum_{v \in N(u)} \sum_{k \in CH} [s_t^k(e_{uv}) + s_t^k(e_{vu})] \leq R(u), \forall u \in V \quad (2)$$

#### 4.3.3.3 Contrainte d'interférence

La capacité d'un WMN est contrainte par l'interférence entre les différents nœuds. Nous mettons en œuvre le modèle d'interférence de protocole [42]. Dans ce modèle, une transmission sur le canal  $I$  et sur la liaison  $e$  est réussie quand tous les émetteurs qui s'interfèrent potentiellement dans le voisinage de l'expéditeur  $t(e)$  et de receveur  $r(e)$  sont silencieux sur le canal  $I$  pour la durée de la transmission. C'est semblable au modèle utilisé dans IEEE 802.11, basé sur un ordre RTS-CTS-DATA-ACK.

$$\sum_{e \in E(t(e)) \cup E(h(e))} s_t^k(e) \leq 1, \forall k \in CH, \forall e \in V, \forall t \quad (3)$$

La figure 4.2 illustre la contrainte d'interférence sur la liaison  $(v, v1)$ . Pour que la transmission sur la liaison  $(v, v1)$  soit réussie, il faut que les nœuds qui sont dans le voisinage de  $v$  et de  $v1$  restent silencieux (les ensembles  $t(e) = \{v2, u, v3\}$ ,  $r(e) = \{v5, v4\}$ ).

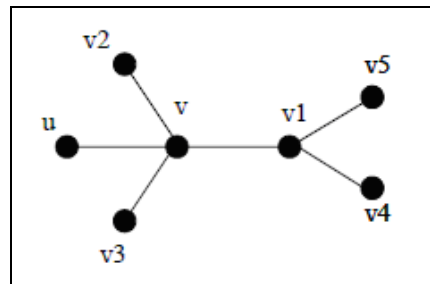


Figure 4.2 : Contrainte d'interférence

Quelques exigences supplémentaires doivent être ajoutées pour modéliser les contraintes de la topologie du réseau et du modèle de différenciation de service.

#### 4.3.3.4 Contraintes de la topologie du réseau

Supposons qu'un flux de trafic du nœud routeur  $x$  vers la passerelle GW est sur un chemin, dénoté par  $P_{(x, GW)}$ . Dans une topologie en arbre, un nœud intermédiaire sur un chemin a seulement un seul nœud suivant. Supposons un nœud  $u$  est un nœud intermédiaire appartenant à  $P_{(x, GW)}$ . Supposons qu'un autre flux de trafic du nœud  $y$  vers la passerelle GW sur le chemin  $P_{(y, GW)}$  a aussi comme nœud intermédiaire, le nœud  $u$ . Puisque le nœud  $u$  a seulement un seul nœud suivant pour l'expédition de trafic, le nœud suivant du nœud  $u$  sur le chemin  $P_{(y, GW)}$  est le même pour le chemin  $P_{(x, GW)}$ . Nous dénotons ce nœud suivant comme  $SUIV(u)$ , le nœud  $u$  reçoit le trafic depuis ces nœuds enfants dans l'arbre et nous les présentons comme  $CL(u)$  et le flux de trafic sur la liaison  $e_{u, v}$  comme  $f(e_{u, v})$ . Tous les flux de trafic dans un WMN forment une topologie à base d'arbre comme les lignes continues montrées dans la figure 4.3.

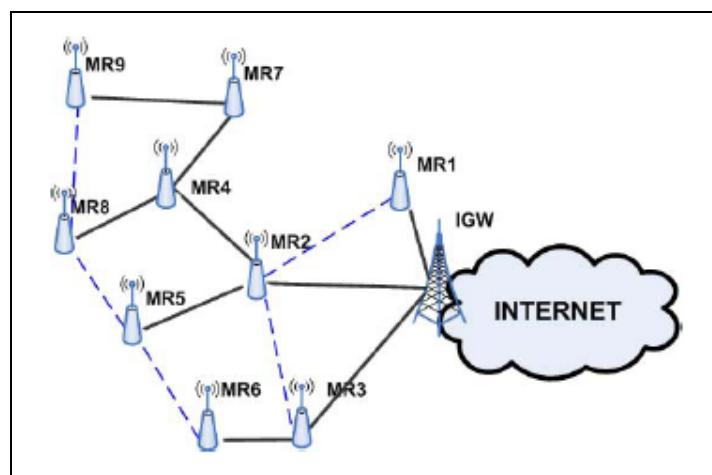


Figure 4.3 : Topologie de WMN en arbre

L'équation (4) illustre le flux de trafic partant du nœud  $u$ , où  $\lambda_u$  dénote le trafic des utilisateurs mobiles associés au nœud  $u$ .

$$f(e_{u, p(u)}) = \lambda_u + \sum_{w \in CL(u)} f(e_{w u}) \quad (4)$$

Particulièrement, pour la passerelle, nous avons la contrainte (5), où  $CR(GW)$  est l'ensemble de canal utilisé par les interfaces radio de GW qui est un sous-ensemble de  $CH$ , et  $W_{e(u, GW)}^k$  est la bande passante du canal  $k$  sur la liaison  $e_{(u, GW)}$ .

$$\sum_{u \in CL(GW)} f(e_{u, GW}) \leq \sum_{k \in CR(GW)} W_{eu, GW}^k \quad (5)$$

Cette équation indique que la bande passante de réseau est contrainte par la capacité de la passerelle GW. L'objectif de l'attribution de canal sera donc de maximiser la capacité de la passerelle pour servir le plus grand nombre de flux dans le réseau.

$$\text{Max} \sum_{k \in CR(GW)} W_{eu, GW}^k \quad (6)$$

#### 4.3.3.5 Contraintes du modèle de différenciation de service

Avec l'approche de différenciation de service, l'objectif est de maximiser l'utilisation des ressources de réseau pour satisfaire le plus grand nombre de demandes avec QoS, mais tout en évitant le problème d'inter-classes (famine). Il faut satisfaire donc le maximum de demandes de la classe BE. Ces objectifs peuvent être formulés comme suit :

##### *Objectif des flux avec QoS*

$$\text{Max} \sum_{u \in CL(GW)} f_{QoS}(e_{u, GW}) \quad (7)$$

$F_{QoS}(e_{u, GW})$  représente le flux de la classe QoS qui passe par la liaison  $e_{u, GW}$ .

##### *Objectif des flux Internet de la classe BE*

$$\text{Max} \sum_{u \in CL(GW)} f_{BE}(e_{u, GW}) \quad (8)$$

$F_{BE}(e_{u, GW})$  représente le flux de la classe Best Effort qui passe par la liaison  $e_{u, GW}$ .

**Mixage entre les différents flux (avec et sans exigence de QoS)**

$$\text{Max} \sum_{u \in \text{CL}(\text{GW})} \left( f_{\text{EF}}(e_{u,\text{GW}}) + f_{\text{BE}}(e_{u,\text{GW}}) \right) \quad (9)$$

**Contrainte de capacité**

Un réseau mesh est représenté par un ensemble de ressources  $E$  ou chaque ressource  $e \in E$  est une paire (lien  $u$ , canal  $c$ ) et chaque lien  $u$  opère sur un canal  $c$  à une capacité finie  $c_e$ . Ce réseau est utilisé par un ensemble de flux  $F$ . Chaque flux  $f$  a une exigence  $x_f$  sur la largeur de bande passante et doit traverser une séquence de ressources (les liens constituent son chemin de bout en bout) pour atteindre sa destination. Soit un booléen  $R_{ef}$  qui indique si la ressource  $e$  est utilisée par le flux  $f$  ou non. La contrainte de ressource est donnée par la formule suivante [44]:

$$\sum_{f \in F} R_{ef} * x_f \leq c_e \quad \forall e \in E \quad (10)$$

Dans l'objectif de maximiser le débit (6) et de servir le plus grand nombre de demandes des deux classes EF et BE (9) dans les contraintes (1), (2), (3), (5) et (10), un problème de Programmation Linéaire en Nombre Entier (PLNE) peut être formé.

Satisfaire toutes les contraintes et les objectifs de (1) - (9) est un problème très complexe car plusieurs variables s'impliquent. De plus, ces variables varient avec le temps, ce qui rend le problème plus dur.

Donc, nous proposons une heuristique d'attribution de canal, qui est basée sur une topologie à base d'arbres et satisfait les différentes contraintes. La solution proposée utilise un multi-arbres, c'est-à-dire un arbre pour chaque classe de trafic.

Nous résumons dans ces tableaux les différentes contraintes et objectifs de notre problème :

**Tableau 4.2 : Table des contraintes de notre problème**

Contrainte	Détaille
Liaison	$\sum_{k \in \text{CH}} s_t^k(e_{uv}) \leq h(e_{uv}), \forall e_{uv} \in E$
Interface radio	$\sum_{V \in N(u)} \sum_{k \in \text{CH}} [s_t^k(e_{uv}) + s_t^k(e_{vu})] \leq R(u), \forall u \in V$
Interférence	$\sum_{é \in E(t(e)) \cup E(h(e))} s_t^k(é) \leq 1, \forall k \in \text{CH}, \forall e \in V, \forall t$
Topologie en arbre	$\sum_{u \in \text{CL}(\text{GW})} f(e_{u,\text{GW}}) \leq \sum_{k \in \text{CR}(\text{GW})} W_{eu,\text{GW}}^k$

Tableau 4.3 : Table des objectifs de notre problème

Objectif	Détaille
Maximiser la capacité de la passerelle	$\text{Max}_{k \in \text{CR}(\text{GW})} \sum W_{eu, \text{GW}}^k$
Satisfaire le Max de demande (EF, BE)	$\text{Max}_{u \in \text{CL}(\text{GW})} \sum (f_{\text{EF}}(e_{u, \text{GW}}) + f_{\text{BE}}(e_{u, \text{GW}}))$

#### 4.4 Algorithme d'attribution de canal

Le problème qu'on doit résoudre est présenté dans la figure suivante.

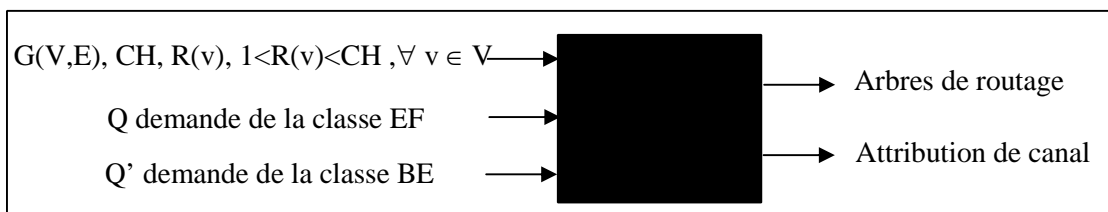


Figure 4.4 : Problème d'optimisation

Pour le résoudre, nous proposons un nouveau mécanisme d'attribution de canal pour le support de QoS avec plusieurs arbres de routage, qui construit des arbres recouvrants basé sur le type de trafic et améliore la qualité de service et l'efficacité d'équilibrage de charge ainsi que la tolérance aux pannes.

##### 4.4.1 Avantage de topologie en arbre

Nous choisissons la topologie en arbres à cause de l'architecture à trois niveaux du WMN où la plupart du trafic est entre les nœuds et l'Internet filaire et où le trafic point à point est quasi nul [38] [26] [30] [11] [34]. Cette topologie offre un mécanisme de routage approprié pour le trafic orienté passerelles (plusieurs sources vers une même destination) [43].

- Les routeurs ne maintiennent pas d'autres informations d'état que de leurs voisins immédiats. Les informations de routage par destinations conventionnelles ne sont pas nécessaires. Les paquets partants d'un nœud suivent simplement l'arbre vers la racine, (le nœud de destination). Les paquets entrants suivent des chemins explicites (routage de source) donnés par les nœuds parents.
- Un avantage supplémentaire de l'utilisation d'un arbre réside dans l'élimination des boucles de routage.

Nous choisissons le concept multi arbres car il vient avec la différenciation de service et l'équilibrage de charge. Quand un seul arbre recouvrant est partagé avec tout les routeurs comme dans [26] par exemple, nous éprouvons une situation de famine de trafic moins prioritaire ou une dégradation de qualité de service. Au contraire, si un arbre recouvrant différent est construit pour chaque type de trafic, ces problèmes peuvent être surmontés mais il faut répondre aux questions suivantes :

- Comment construire ces différents arbres ?
- Comment partager les ressources du réseau sur ces arbres de routage ?
- Etant donné que le nombre d'interfaces radio est limité, Comment attribuer les canaux aux interfaces radio des nœuds du réseau pour assurer la connectivité des différents arbres?
- Comment peut on assurer un équilibrage de charge sur les deux arbres construits ?

Dans ce qui suit, nous essayons de donner des réponses à toutes ces questions.

#### **4.4.2 Construction des arbres de routage et attribution de canal**

Notre solution inclut deux phases.

1. Dans la première phase, la passerelle découvre la topologie à base d'arbres.

2. Dans la deuxième phase, les canaux vont être assignés niveau par niveau en commençant par la passerelle. Nous disons un niveau pour référer au nombre de sauts entre la passerelle et un nœud.

Une fois que les deux phases sont exécutées, on garde l'attribution pour une certaine durée à moins qu'une procédure de réattribution ne soit déclenchée, s'adaptant ainsi au changement de topologie ou de charge du réseau. Le détail de ces processus est comme suit :

##### **4.4.2.1 Construction des arbres de routage**

Pour arriver à notre objective, l'idéal est d'avoir des chemins différents pour chaque classe de trafic. Il faut donc construire deux arbres de routage, le premier arbre pour router les flux avec exigences de QoS (donc cet arbre doit avoir le maximum de bande passante) et le deuxième arbre pour transporter les autres flux du réseau (sans exigence de QoS). Cela dépend des ressources du réseau disponibles (nombre d'interface radio dans chaque nœud et nombre de canaux dans le réseau).

##### **a) Détection du voisinage et problème de diffusion**

Dans une phase d'initialisation, pour détecter la topologie physique du réseau, chaque nœud dans le réseau commute une de ses interfaces radio vers un canal commun et diffuse périodiquement un message 'Hello' portant son identificateur ID pour informer ses voisins de sa présence dans le réseau et pouvoir recueillir la liste des voisins directs. Après cette phase

qui dure pendant une période prédéfinie, chaque nœud possède la liste de ses voisins directs (à un saut) et à deux sauts.

Le nœud passerelle entame, par la suite, le processus de construction des arbres de routage, puis entame le processus d'attribution de canal (ses deux processus sont détaillés dans les sections suivantes).

Après l'attribution de canal, la diffusion dans le réseau devient un problème car les nœuds du réseau multi-canaux multi-radios ne partagent aucun canal commun avec leurs voisins. Deux solutions ont été proposées.

- La première solution proposée dans [34] [25] consiste, tout simplement, à consacrer une interface radio dans chaque nœud et l'opérer sur un canal commun spécialement pour échanger les messages de diffusion.
- Raniwala et Al dans [26] présente une solution plus intéressante nommée, '*virtual control network* (contrôle virtuel de réseau)' qui vise à sauver une interface radio dans chaque nœud par le mécanisme suivant : Puisqu'il y a toujours un chemin entre un nœud et ses voisins physiques. Un message de contrôle peut être livré à travers un ou plusieurs sauts sur le réseau mesh. Pour des raisons d'efficacité, la diffusion des messages de contrôle est assurée par l'utilisation de multicast IP. L'idée de base est de mettre en œuvre les communications de la couche 2 en utilisant le routage de la couche 3. Avec un contrôle virtuel, un nouveau nœud doit parcourir tous les canaux pour diffuser un message Hello pendant la phase de découverte de voisin. le processus de balayage de canal peut être fait dans 5 à 10 millisecondes. Quand deux nœuds parcourent les canaux simultanément, chaque nœud utilise seulement son interface radio montant pour exécuter le balayage, en gardant ses interfaces radios descendantes à des canaux fixes. Cela assure que les nœuds voisins peuvent finalement se découvrir.

Avec l'objectif de maximiser la capacité du réseau, le nombre d'interfaces radio et le nombre de canaux sont des ressources critiques, qu'on doit les exploiter de manière optimale. La première solution s'avère donc inintéressante. La deuxième solution présente l'inconvénient de temps surtout pour les flux qui exige des délais minimums (flux QoS).

Pour implémenter la diffusion d'un message, nous allons adopter la deuxième solution avec des petites modifications. La même idée de balayage des canaux va être appliquée, mais un nœud donné utilise seulement l'arbre BE pour l'implémenter. Pour minimiser le temps de balayage, il faut réduire le nombre de canaux qui doivent être scannés. Pour cela, chaque nœud doit maintenir une table de canaux du voisinage. Cette table a autant d'entrées que de

nœud voisin direct et chaque entrée contient un ou plusieurs identificateurs de canaux selon le nombre d'interfaces radio du nœud voisin.

### b) Processus de construction des arbres de routage

Le processus de construction des arbres de routage reçoit en entrée les deux vecteurs de charge de trafic et le nombre d'interfaces radio et de voisin de chaque nœud, et fournit en sortie les arbres de routage comme montre la figure suivante.

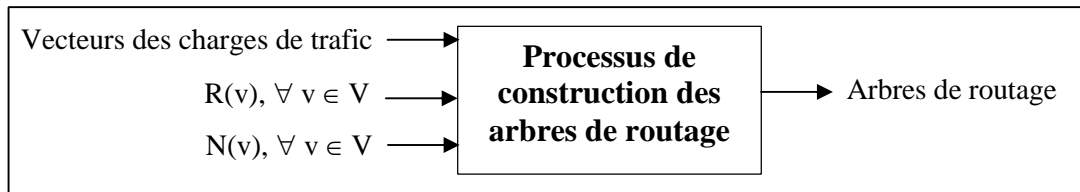


Figure 4.5 : Entrés/Sorties du processus de construction des arbres de routage

Pour construire les arbres de routage, la passerelle diffuse périodiquement un message TR-CONST (TRee CONSTruction) vers ses voisins. Ce message comprend les informations suivantes:

- Nombre d'interfaces radio du nœud transmetteur (le nœud qui a plus d'interface radio aura plus de chance d'être choisi comme parent).
- Quantité de flux: les prévisions de charge passant par le nœud transmetteur ( $CH_{EF}$ ,  $CH_{BE}$ ). Ces prévisions sont obtenues à partir du modèle de trafic du réseau mesh qui est calculé en fonction de nombre de flux ou de chemins qui passent par ce nœud comme dans [25].
- Nombre de voisins du nœud transmetteur qui peuvent être des enfants par la suite (le nœud qui a moins de voisins aura plus de chance d'être choisi comme parent).
- Distance : nombre de sauts entre le nœud et la passerelle (elle sert à choisir des chemins courts avec moins de délai en l'absence d'interférences). Ce nombre sera incrémenté à chaque expédition.
- Numéro de séquence : un identificateur unique pour chaque message TR-CONST pour éviter des messages doubles.

Pour les autres nœuds, après le démarrage, un nœud reste en silence, attend des messages (TR-CONST) de tous ses voisins. A la réception de ce message, un nœud ajoute l'expéditeur à son ensemble de parents de secours, et calcule la métrique de routage en fonction des informations trouvées dans ces messages reçus et choisit le meilleur voisin comme son parent pour chaque arbre. Il envoie par la suite un message JOINT pour une demande de jointure aux voisins choisis pour être un membre de l'arbre de routage associé. Le calcul de la métrique de

Le routage des deux arbres se différencie selon la charge de trafic de chaque classe (EF et BE). Si la demande de jointure est acceptée, le nœud parent insère ce nœud dans sa liste d'enfants de l'arbre en question, met à jour ses tables de routage et informe ses nœuds parents de la jointure d'un nouveau nœud par l'envoi récursif du message CH-ADD qui porte comme paramètre l'identificateur du nœud nouvellement associé. Ils répondent aussi par le message ACCEPTE à la demande de jointure pour que ce nœud diffuse itérativement le message TR-CONST et le processus de construction des arbres poursuit jusqu'à l'arrivée au dernier niveau où un nœud n'a pas de voisin supplémentaire.

Le message TR-CONST va être transmis périodiquement pour équilibrer la charge de trafic dans les arbres de routage et pouvoir détecter d'éventuelles défaillances dans le réseau.

Chaque nœud maintient une entrée dans les deux tables de routage pour chacun de ses nœuds descendants. Par exemple, un nœud passerelle garde une entrée dans chaque table de routage pour chaque nœud dans son arbre recouvrant associé. Cependant, un nœud ne doit pas maintenir des entrées de table de routage pour ses nœuds parents. Au lieu de cela, chaque nœud utilise son parent comme routeur par défaut, c'est-à-dire les paquets dont la destination n'est pas un de ses nœuds descendants sont simplement expédiés à son nœud parent suivant leurs classes. Par conséquent, un nœud peut atteindre ses ancêtres par le biais de son nœud parent par défaut, et peut atteindre ses nœuds frères en utilisant une route triangulaire par l'intermédiaire de leur nœud ancêtre commun.

En résumé, étant donné un ensemble de nœuds sans fil, notre algorithme de routage permet à chaque nœud de participer à un arbre de routage pour chaque classe de trafic. Un nœud peut avoir donc plus d'un parent selon leurs charges de trafic. Pour cela, chaque nœud a autant de tables de routage que de classes définies. Chaque table décrit le sous arbre associé au nœud

Le tableau suivant détaille tous les messages utilisés dans la construction de l'arbre de routage.

**Tableau 4.4 : Table des messages utilisés dans la construction de l'arbre de routage**

Message	Détaille
TR-CONST	Envoyé périodiquement pour la construction de l'arbre de routage, l'équilibrage de charge et la détection de panne
JOINT	Envoyé comme réponse au message TR-CONST pour demander la jointure au sous-arbre du destinataire.
ACCEPT	Réponse positive au message JOINT.
CH-ADD	Envoyé au nœud parent après jointure d'un nouveau nœud enfants.

Le diagramme suivant montre le processus de construction des arbres de routage.

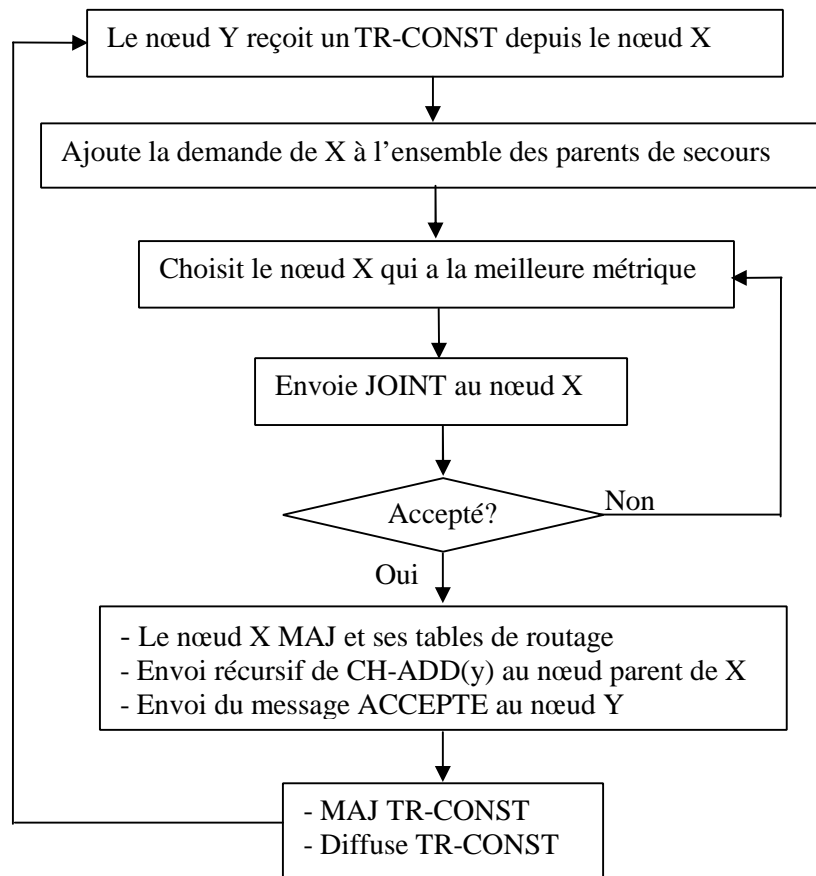


Figure 4.06 : Processus de construction des arbres de routage

### c) Exemple

Soit le réseau décrit dans la figure 4.7 où chaque nœud est représenté par un vecteur de deux champs (charge de trafic de flux QoS, charge de trafic de flux BE). Le nœud 'A' est considéré comme passerelle. Il est donc la racine des deux arbres de routage. Cette figure montre les différents arbres de routages construits (les lignes continus représentent les arcs de l'arbre QoS, les lignes discontinus désignent les arcs de l'arbre BE).

Plusieurs facteurs influencent sur la construction des différents arbres tels que la distribution des nœuds dans le réseau, l'emplacement des nœuds passerelles et les charges de trafic transitant par chaque nœud.

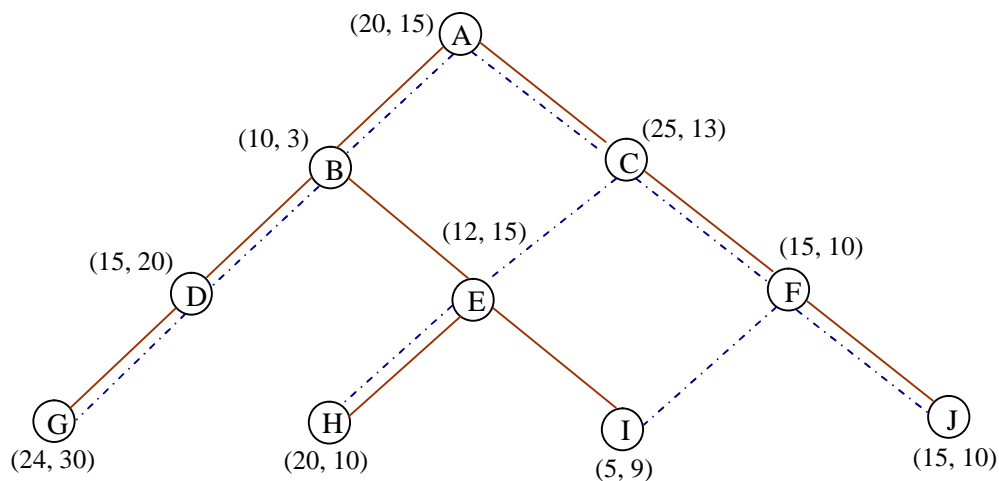


Figure 4.7 : Exemple d'arbres de routage

#### d) Métrique de routage

Le but principal de la construction des arbres de routage est de trouver le meilleur chemin entre chaque nœud et la passerelle pour chaque classe de trafic. La construction des arbres de routage est basée sur une fonction qui prend en considération : le nombre d'interfaces radio dans un nœud, la charge de trafic, le nombre de sauts à partir de la passerelle et le nombre de voisins.

La métrique de routage d'un nœud détermine sa chance d'être accepté comme nœud parent dans la construction des arbres de routage. Nous proposons la métrique suivante qui améliore celle proposée par Raniwala dans [26] et l'autre proposée dans [33] par l'ajout du nombre de voisins directs d'un nœud.

$$\text{Métrique} = \frac{\text{Nb-interfaces-nœud}}{\text{Charge-traffic} * \text{Distance (passerelle, nœud)} * \text{Nb-voisins}} \quad (11)$$

- Le nombre d'interfaces radios d'un nœud affecte proportionnellement sa priorité d'être choisi comme parent dans l'arbre de routage (c'est-à-dire, plus grand est le nombre d'interfaces radios, plus élevée sera la priorité dans la construction des arbres de routages). Si un nœud a plusieurs interfaces radios, ses enfants par la suite vont avoir plus de chance d'utiliser une interface, donc un canal individuellement, ce qui diminue l'interférence inter-chemin. Ainsi, le nombre d'interfaces radio dans le numérateur dans l'équation (11) augmente la métrique d'un nœud avec plusieurs interfaces radios.
- Il est clair que la charge du trafic passant à travers d'un nœud a un impact sur le calcul du meilleur chemin. Le nœud qui relaie moins de trafic, va être choisi comme parent, pour

avoir un chemin avec plus de débit. Ainsi, la charge de trafic dans le dénominateur dans l'équation (10) diminue la métrique d'un nœud avec plus de trafic.

- Aussi, en raison de la nature hiérarchique de la topologie mesh, les nœuds qui sont plus proches de la passerelle doivent être choisis avec priorité dans le calcul des chemins pour avoir des chemins avec un délai minimal.
- En même temps, le nombre de voisins directs d'un nœud qui peuvent être des nœuds enfants dans l'arbre de routage affecte inversement sa priorité d'être choisi comme parent dans l'arbre de routage, car il va augmenter l'interférence inter-chemin par la suite.

#### e) Estimation de charge de trafic d'un lien ou d'un nœud

L'estimation de la charge de trafic doit indiquer la charge de trafic de chaque classe. Comme il existe deux types de classes, à savoir, QoS et BE, nous présentons la charge de trafic par un vecteur de deux champs, un champ pour chaque classe. Nous pouvons différencier entre les flux des différentes classes par un champs spécial dans l'entête de paquet qui indique la classe du flux. Pour mesurer la charge de trafic d'un nœud nous calculons le nombre de flux qui passent par un nœud comme dans [25].

#### 4.4.2.2 Attribution de canal

##### a) Partage des interfaces radio entre les deux arbres

Pour éviter le problème d'inter-classes (famine), les protocoles de routages dans les réseaux filaires visent à réserver une partie de la bande passante disponible dans le réseau pour les flux avec exigences de QoS et utilise l'autre partie pour servir les flux BE [45]. Si on note la bande passante disponible dans le réseau par  $BP_{disp}$ , la bande passante réservée pour les flux avec QoS par  $BP_{QoS}$ , et la bande passante des flux de BE par  $BP_{BE}$ . Alors nous aurons la relation suivante :

$$\begin{cases} BP_{QoS} = \lambda * BP_{disp} \\ BP_{BE} = (1 - \lambda) * BP_{disp} \end{cases} \quad 0 < \lambda < 1 \quad (12)$$

$\lambda$  représente alors le pourcentage de la bande passante disponible réservé pour les flux QoS.

En suivant le même principe pour partager le nombre d'interfaces radio de chaque nœud sur les deux arbres de routages, le paramètre  $\lambda$  sera calculé sur la base des charges de trafic des deux classes.

- La charge de trafic d'une classe donnée est calculée en fonction du nombre de flux.

- Un flux de la classe QoS demande plus de ressources par rapport à un flux de la classe BE, on peut écrire alors l'équation suivante :

$$\text{Flux (QoS)} = k * \text{Flux (BE)} \quad (13)$$

- Le paramètre  $k$  est un nombre entier supérieur à 1. Il représente le facteur de la bande passante demandé par un flux QoS par rapport à un flux BE.

$\lambda$  est donné alors par la formule suivante

$$\lambda = \text{NB-Flux (QoS)} / (\text{NB-Flux (BE)} + \text{NB-Flux (QoS)}) \quad (14)$$

Le nombre d'interfaces radio qui sera attribué pour chaque arbre de routage dans chaque nœud  $v$  est donné par la relation suivante

$$\begin{cases} 2 \leq \text{NB-Radio}_{\text{QoS}}(v) \leq \lambda * R(v) \\ 2 \leq \text{NB-Radio}_{\text{BE}}(v) \leq (1-\lambda) * R(v) \end{cases} \quad \forall v \in V \quad (15)$$

Nous exigeons que la valeur minimale du nombre d'interfaces radio dédié pour chaque arbre soit égale à deux car l'interface radio qui relie un nœud avec son parent doit être différente de celles reliant le nœud avec ses enfants, pour minimiser l'impact du problème de dépendance de canaux dans le cas de réattribution de canal.

Cette contrainte peut être vue comme un inconvénient : une réservation de ressource qui peut ne pas être utilisée car nous pouvons trouver qu'un nœud (client) qui sollicitera des connexions dans une seule classe. Mais cette situation est rare parce qu'un nœud routeur est considéré comme une station de base pour un ensemble de clients; alors la probabilité que tous ses clients demandent une connexion dans une seule classe est faible. Un nœud routeur a donc du trafic permanent dans les deux classes.

#### **b) Partage des canaux entre les deux arbres**

Les canaux disponibles dans le réseau seront partagés en respectant le système de priorité suivant : pour router les flux de type QoS, un nœud peut utiliser les meilleurs canaux disponibles dans les réseaux et le reste des canaux sera utilisé pour les flux de type BE. Ce sous ensemble est défini selon le paramètre  $\lambda$  défini ci-dessus. Le nombre de canaux qui sera attribué pour chaque arbre de routage dans chaque nœud  $v$  est donné donc par la relation suivante :

$$\begin{cases} \text{NB-Canaux}_{\text{EF}}(v) = \lambda * \text{CH} \\ \text{NB-Canaux}_{\text{BE}}(v) = (1-\lambda) * \text{CH} \end{cases} \quad \forall v \in V \quad (16)$$

Dans le processus d'attribution de canal, chaque nœud commence par l'attribution de canal aux interfaces radios de l'arbre QoS d'abord, puis il utilise les autres canaux pour les attribuer aux interfaces radios de l'arbre BE. Puisque le nombre d'interfaces radios d'un nœud donné dédié pour chaque arbre est toujours inférieur au nombre de canaux disponibles dans le réseau, nous pouvons tirer les conclusions suivantes :

- Chaque nœud utilise les meilleurs canaux disponibles (un sous ensemble) pour les attribuer aux interfaces radios de l'arbre QoS.
- Les interfaces radios d'un nœud donné ne partagent aucun canal commun. (c'est-à-dire que les interfaces radios de l'arbre BE ne peuvent pas utiliser les canaux attribués aux interfaces radios de l'arbre QoS dans un nœud donné)
- Par contre, deux nœuds voisins physiques, qui ne sont pas liés logiquement, peuvent partager des canaux communs entre leurs deux types d'interfaces radio suivant l'état d'utilisation des canaux.

### c) Estimation de charge des canaux

L'information de charge des canaux est donnée par l'envoi périodique d'un message spécial 'CH-USAGE'. Chaque nœud mesure sa propre utilisation des différents canaux par ses différentes interfaces radios et envoie périodiquement ce message vers ses voisins dans sa zone d'interférence. Ce message contient les paramètres de deux listes dont chacune est constituée d'un ensemble de paires (canal, charge). Chaque paire contient deux sortes d'information : une qui décrit la charge des canaux utilisés par les interfaces radio QoS et une autre qui donne la charge des canaux attribués aux interfaces radios BE. Cette information aide les nœuds de choisir les canaux les moins utilisés (moins chargés) dans la phase d'attribution de canaux aux interfaces, pour équilibrer les charges des canaux.

Chaque nœud maintient alors deux matrices de charge des canaux et chaque matrice reflète l'état d'utilisation des canaux par les différents types d'interfaces radio,  $M_{\text{QoS}}(Z, \text{CH})$ ,  $M_{\text{BE}}(Z, \text{CH})$ , de telle sorte que  $Z$  représente le nombre de ses voisins dans la zone d'interférence et  $\text{CH}$  est le nombre de canaux disponibles.

Pour augmenter la capacité du réseau (objectif (6)), dans l'estimation de la charge d'un canal, nous devons prendre en considération la priorité d'un nœud voisin avec l'état d'utilisation de canal. La priorité d'un nœud est proportionnelle à sa distance de la passerelle. Le nœud  $N$

calcule alors la charge  $\Phi_c$  d'un canal  $c$ , dans sa zone d'interférence  $Z_{\text{interf}}(N)$ , par la formule suivante :

$$\Phi_c = \sum_{u \in Z_{\text{interf}}(N)} \frac{K * M_{\text{QoS}}(u,c) + M_{\text{BE}}(u,c)}{\text{Priorité}(u)} \quad (17)$$

Le paramètre  $K$  est le même que celui utilisé dans l'équation (13). Il représente le facteur de la bande passante demandé par un flux QoS par rapport à un flux BE

Cette formule permet de choisir le canal le moins utilisé dans la zone d'interférence par le nœud le moins prioritaire. Elle privilège le canal le moins utilisé par le trafic BE sur celui occupé par le trafic QoS.

#### d) Processus d'attribution de canal dans chaque arbre

Puisque chaque nœud dans le réseau n'utilise aucune ressource en commun entre ses deux arbres de routage, le processus d'attribution de canal est le même pour chaque arbre. Ce processus reçoit en entrée l'arbre de routage, le graphe d'interférence, le nombre d'interfaces radio et le nombre de canaux disponible pour chaque nœud et fournit en sortie le même arbre mais avec des canaux assignés comme le montre la figure suivante :

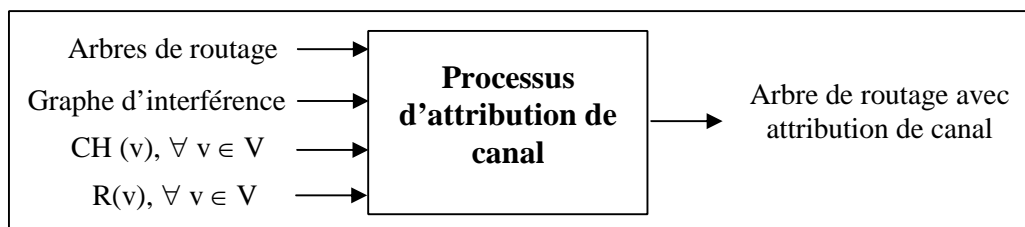


Figure 4.8 : Entrés/Sorties du processus d'attribution de canal

Après que la passerelle construit les arbres de routage, elle commence l'attribution de canal. En général, le lien qui a plus de trafic devrait avoir la plus haute priorité d'avoir le canal qui a la plus grande bande passante. Pour un arbre donné, prenant un nœud  $u$  dont son parent est  $p(u)$  et ses enfants sont  $CL(u)$  et le trafic  $f(e_{(u, p(u))})$  est plus grand que  $\sum_{w \in CL(u)} f(e_{(u, w)})$  selon l'équation (4) la liaison entre le nœud  $u$  avec son parent  $p(u)$  devrait avoir une priorité plus haute que les liaisons entre le nœud  $u$  et ses enfants  $CL(u)$ . L'attribution de canal devrait commencer par la passerelle dans l'arbre.

Pour les liaisons qui relient la passerelle  $GW$  avec ses enfants  $CL$ ,  $e_{u, GW}$ ,  $u \in CL(GW)$ , on devrait assigner le canal qui a la plus grande bande passante pour le lien ayant la plus grande charge de trafic. Supposons que les liaisons soient triées par ordre décroissant de flux de

trafic:  $f_{EF}(e_{u1, IGW}) \geq f_{EF}(e_{u2, IGW}) \geq \dots$ , alors la passerelle GW assigne les canaux selon cet ordre. Si le nombre d'enfants de la passerelle est inférieur ou égal au nombre de ses interfaces radios ( $|CL(GW)| \leq R(GW)$ ), il y a donc assez d'interfaces radios pour que chaque liaison aie un canal différent. Sinon, quelques nœuds enfants vont partager certains canaux entre eux. Considérant la situation suivante, après que la passerelle GW aie assigné  $R(GW)$  canaux pour  $R(GW)$  de ses enfants, le reste d'enfants ( $CM = CL(GW) - R(GW)$ ) doivent partager  $CM$  canaux précédemment assignés parce qu'aucune nouvelle interface radio n'est disponible pour un nouveau canal.

Pour le choix de canal à partager et à assigner dans  $CM(GW)$ , la passerelle trie les  $R$  canaux déjà affecté pour ses enfants dans un ordre croissant de leur état d'utilisation, soit  $CT(c)$  désigne la charge de trafic total du canal  $c$ , alors l'ordre sera :  $CT(c1) \leq CT(c2) \leq \dots \leq CT(cR)$ . Donc, il choisit le canal le moins utilisé pour le nœud suivant quand toutes les interfaces radios sont assignées. Alors, pour la  $R(GW) + 2^{\text{ème}}$  liaison dans l'ordre, la passerelle GW choisit le nouveau canal le moins utilisé. Elle continue jusqu'à ce que la dernière liaison dans l'ordre soit assignée.

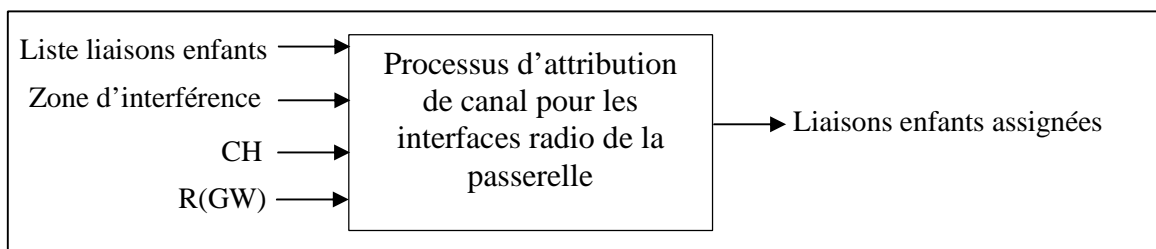
La passerelle informe ses enfants de la décision de l'attribution en envoyant le message CID-ASS (c'est-à-dire, CID ASSignation) à chacun de ses enfants (ce message contient l'identificateur de canal attribué pour cet enfant). Par la suite, chaque enfant répond par un message CID-ACK en tournant son interface radio sur ce canal.

Le tableau suivant présente les messages échangés dans le processus d'attribution de canal

**Tableau 4.5 : Table des messages échangés dans l'attribution de canal**

Message	Détail
CID-ASS	Envoyé aux nœuds enfants après attribution de canal.
CID-ACK	Envoyé au nœud parent comme réponse au message CID-ASS.

Nous résumons dans la figure suivante, les entrées/sorties du processus d'attribution des canaux pour les interfaces radio de la passerelle



**Figure 4.09 : Les E/S du processus CA pour les interfaces radio de la passerelle**

Le diagramme suivant explique le processus d'attribution des canaux pour les interfaces radio de la passerelle.

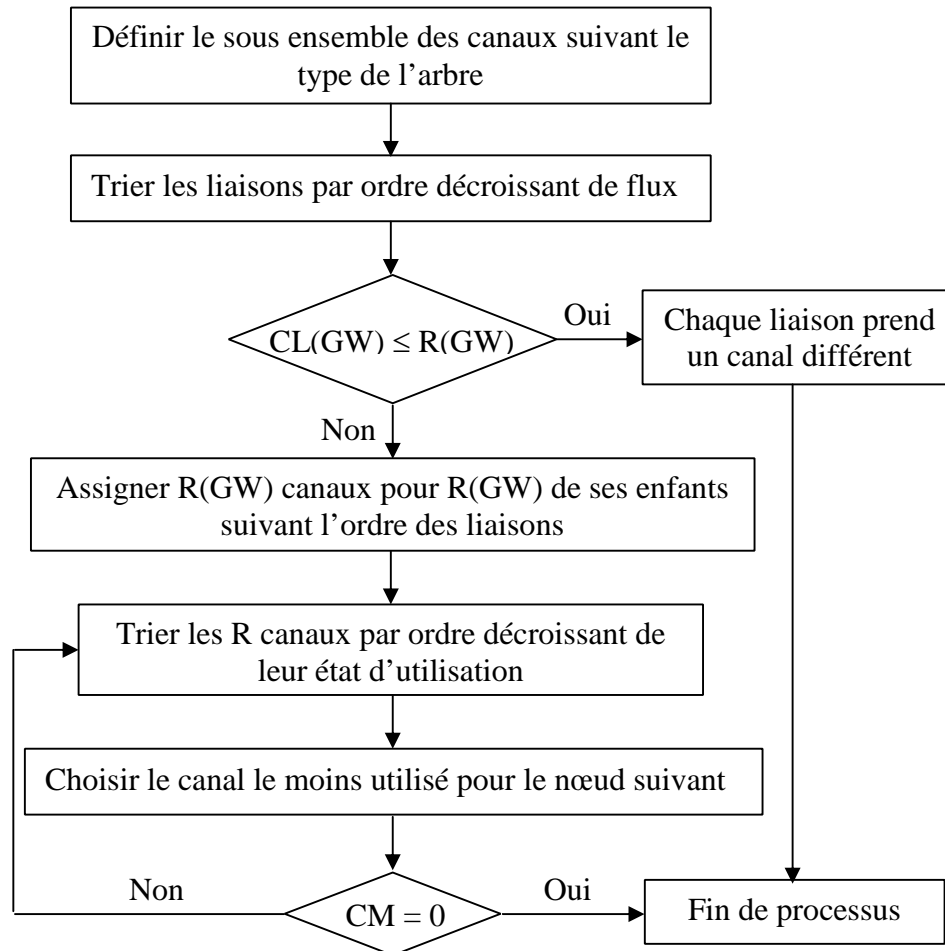


Figure 4.010 : Processus d'attribution des canaux pour la passerelle

Niveau par niveau, chaque nœud assigne les canaux pour son sous arbre. Considérant un nœud  $u$  dans l'arbre. Le nœud  $u$  est au niveau  $i$  et son nombre d'interfaces radios est  $R(u)$ , ses enfants sont  $CL(u)$  et son parent est  $p(u)$ . La différence dans le processus d'attribution de canal entre un nœud intermédiaire et la passerelle est que, avant que  $u$  assigne les canaux pour son sous arbre, il a déjà assigné un canal pour son parent  $p(u)$ . Il doit donc assigner  $CH(u) - 1$  canal disponible pour les  $(R(u) - 1)$  interfaces radios restantes, chaque nœud est responsable de l'attribution de canal sur ces interfaces radios descendants seulement (qui le relie avec ses enfants). Pour faire l'attribution de canal, le nœud  $u$  partage ses canaux disponibles sur trois ensembles.

1. Le premier ensemble ( $CH1$ ) contient les canaux qui ne sont pas utilisés dans la zone d'interférence.

2. Le deuxième ensemble (CH2) inclut les canaux utilisés dans la zone d'interférence moins ceux assignés aux liaisons qui relient  $p(u)$  à ses descendants pour minimiser l'interférence intra-chemin.
3. Le dernier ensemble (CH3) comporte les canaux assignés aux liaisons qui relient  $p(u)$  à ses descendants dans la zone d'interférence.

Avant de commencer l'attribution, ces ensembles doivent être triés par ordre croissant de leur état d'utilisation dans la zone d'interférence.

Si le premier ensemble de canaux (CH1) est suffisant pour l'attribution (c'est à dire  $CH1 \geq R(u)-1$ ), les liaisons entre les enfants  $CL(u)$  et le nœud  $u$ , ne vont pas partager de canal avec leurs voisins dans la zone d'interférence. Autrement, le nœud  $u$  doit choisir convenablement un canal du ensemble (CH2) à partager. S'il reste encore des interfaces non assignées, il utilise le dernier ensemble (CH3) jusqu'à ce que toutes ses interfaces radios soient assignées.

Si le nombre d'enfants d'un nœud est inférieur ou égal au nombre d'interfaces radios restantes ( $CL(u) \leq R(u)-1$ ), chaque interface radio peut être assignée à un enfant  $v$  ( $v \in CL(u)$ ) avec un canal différent. Autrement, quelques enfants doivent partager certains canaux entre eux. La méthode de sélection de canal dans ce cas est semblable avec celle utilisée par la passerelle. Le principe consiste à trouver le canal le moins utilisé parmi les  $R(u) - 1$  canaux assignés à ses interfaces radios d'enfants.

A la fin de ce processus, le nœud  $u$  informe ses enfants de la décision de l'attribution en envoyant le message CID-ASS (c'est-à-dire, CID ASSignation) à chacun de ses enfants. Par la suite, chaque enfant répond par un message CID-ACK en tournant son interface radio sur ce canal.

Nous présentons dans la figure suivante, les entrées/sorties du processus d'attribution des canaux pour les interfaces radio d'un nœud intermédiaire

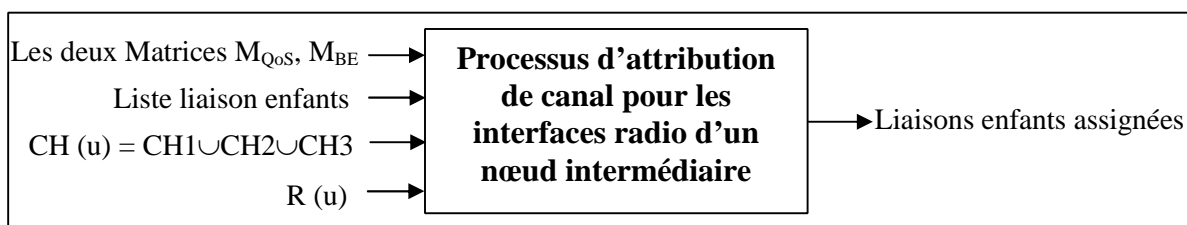


Figure 4.11 : Les e/s du processus CA pour les interfaces radio d'un nœud intermédiaire

Le diagramme suivant détaille le processus d'attribution des canaux pour les interfaces radio d'un nœud intermédiaire.

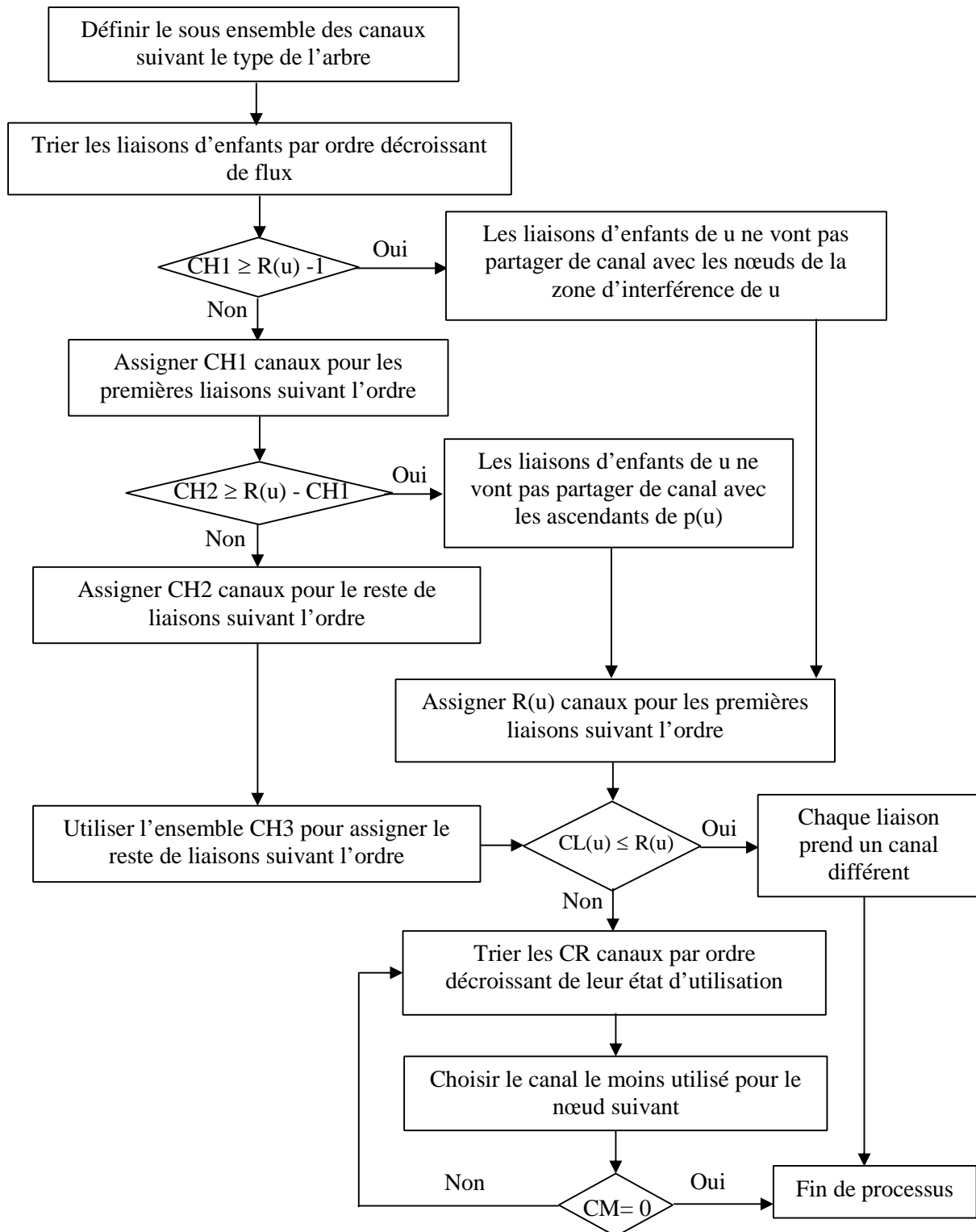
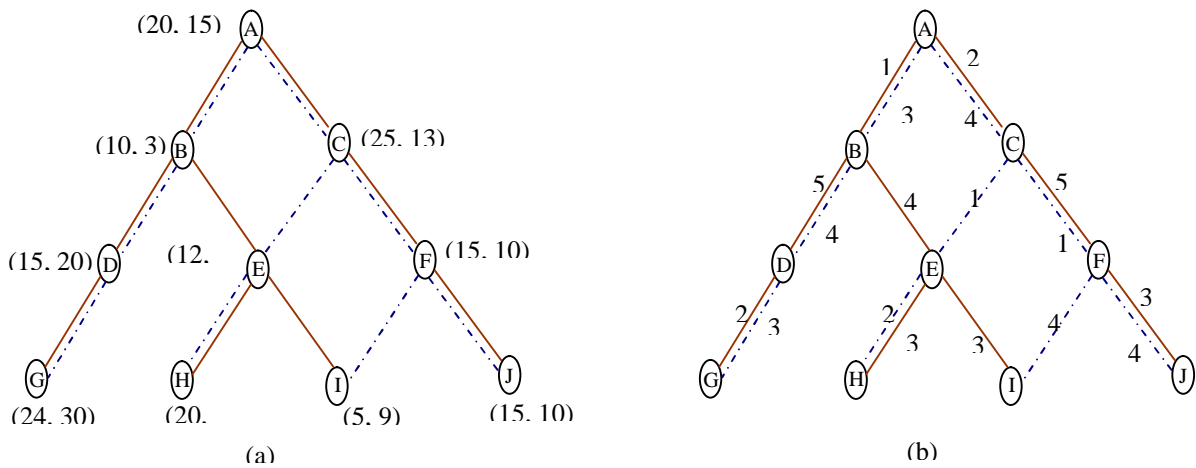


Figure 4.12 : Processus d'attribution des canaux pour un nœud intermédiaire

**e) Exemple**

Nous donnons un exemple pour montrer la procédure d'attribution de canal dans les arbres de routage. Considérons l'exemple précédemment donné dans la construction des arbres de routage. Dans la figure 4.13 (a), nous supposons que chaque nœud a quatre interfaces radios et le nombre total de canaux est 5. On donne à chaque canal un Identification de Canal (CID) nombre (1, 2, 3, 4, 5 ∈ CH). La passerelle A, à deux enfants dans chaque arbre, elle assigne alors, à chaque enfant un canal différent (1, 2, 3, 4) comme montre la figure 4.13 (b).

Après qu'un nœud est assigné à un canal pour sa liaison avec son parent, il lance l'attribution de canal pour son sous arbre. Le processus est semblable aux étapes énumérées dans le cas de la passerelle. En considérant la figure 4.13 (a) comme un exemple, B commute deux interfaces radio sur les canaux 1 et 3 pour ses liaisons avec la passerelle. Une option est possible pour l'interface radio QoS restante: canal 5 qui n'est jamais utilisé (élément de l'ensemble CH1 comme montre le tableau 4.5). Pour attribuer un canal à l'interface radio BE restante, le nœud B, choisit entre les canaux 2 et 4. Supposons qu'il choisit le canal 4 selon son état d'utilisation (moins chargé). Niveau par le niveau, l'attribution de canal est exécutée jusqu'à ce que les interfaces de tous les nœuds soient assignées à un canal comme le montre la figure 4.13 (b).



**Figure 4.13 : Exemple d'attribution de canal**

**Tableau 4.6 : Liste des canaux triés dans chaque ensemble pour chaque nœud**

Ensemble Nœud	CH1		CH2		CH3	
	EF	BE	EF	BE	EF	BE
B	5	5	4, 2	4, 2	φ	φ
C	5	5	1, 3	1, 3	φ	φ
D	2	2	3	1	1	3
E	φ	φ	3, 4, 2	2, 3	φ	4
F	3	3	4	2	2	4

### 4.4.3 Equilibrage de charge

Comme [38] [26], pour maximiser l'utilisation efficace des ressources radio précieuses, il est essentiel que les charges sur chaque nœud soient équilibrées. La construction des arbres de routage et l'attribution de canal dépendent, toutes les deux, de la charge des nœuds et des liaisons. Ces charges sont dynamiques et varient avec le temps. L'équilibrage de charge affecte alors les deux processus, ce qui mène à une reconstruction des arbres de routage ou à une réattribution des canaux.

#### 4.4.3.1 Equilibrage de charge avec réattribution de canal

Pour un meilleur équilibrage de charge, l'attribution de canal doit être ajustée périodiquement. Dans le cas d'une seule classe de trafic, Raniwala dans [26], propose deux phases pour équilibrer les charges des canaux.

a) *Phase d'équilibrage de charge de canaux,*

Dans cette phase, chaque nœud évalue son attribution de canal actuelle basée sur les informations d'utilisation de canal qu'il reçoit à partir de ses nœuds voisins. Quand il trouve un canal relativement moins chargé, il commute un de ses interfaces radios enfant fonctionnant sur le canal le plus chargé pour l'utiliser et envoyer un message de CHANGE-CH-ASS (changement d'assignation de canal) avec les nouvelles informations de canal aux nœuds enfants affectés afin que ces derniers modifient le canal de leur interface radio parent en conséquence. Le nœud envoie aussi le message CHANGE-CH-USAGE, qui porte sur la mise à jour de l'état d'utilisation de canal à ses voisins de sa zone d'interférence pour assurer que d'autres nœuds dans le voisinage ne migrent pas au nouveau canal parce qu'ils supposent (incorrectement) qu'il est toujours moins chargé. Pour réduire la probabilité de commutation simultanée, les phases d'équilibrage de charge entre les nœuds voisins vont être différenciés.

b) *Phase d'équilibrage de charge des interfaces enfants,*

Dans le cas où relativement aucun canal moins chargé n'est disponible pour l'interface radio exploitant le canal le plus chargé, le nœud passe à un équilibrage de charge interne (entre les interfaces enfants). Ici, il redistribue ses enfants sur ses interfaces pour les rendre plus uniformément chargées.

Si les canaux restent encore déséquilibrés, nous ajoutons une troisième phase, que l'on appelle 'Phase de réinitialisation'.

c) *Phase de réinitialisation,*

Dans cette phase, un partage de ressources réseau entre les deux classes sera réexécuté. Puisque le nombre d'interfaces radios dans un nœud est fractionné selon les charges de trafic

de chaque classe dans un nœud donné, alors quand ces charges changent, le partage de ressource doit être recalculé. Le nœud recalcule ici le pourcentage de ressource qu'il doit affecter pour chaque classe. Si le résultat de partage est différent de celui utilisé actuellement, le nombre d'interface radio et de canal affecté pour chaque classe doit être ajusté et une réattribution de canal sur ses deux nouveaux ensembles d'interfaces radio descendants doit être faite par la suite. Sinon, une reconstruction des arbres de routage s'avère donc nécessaire.

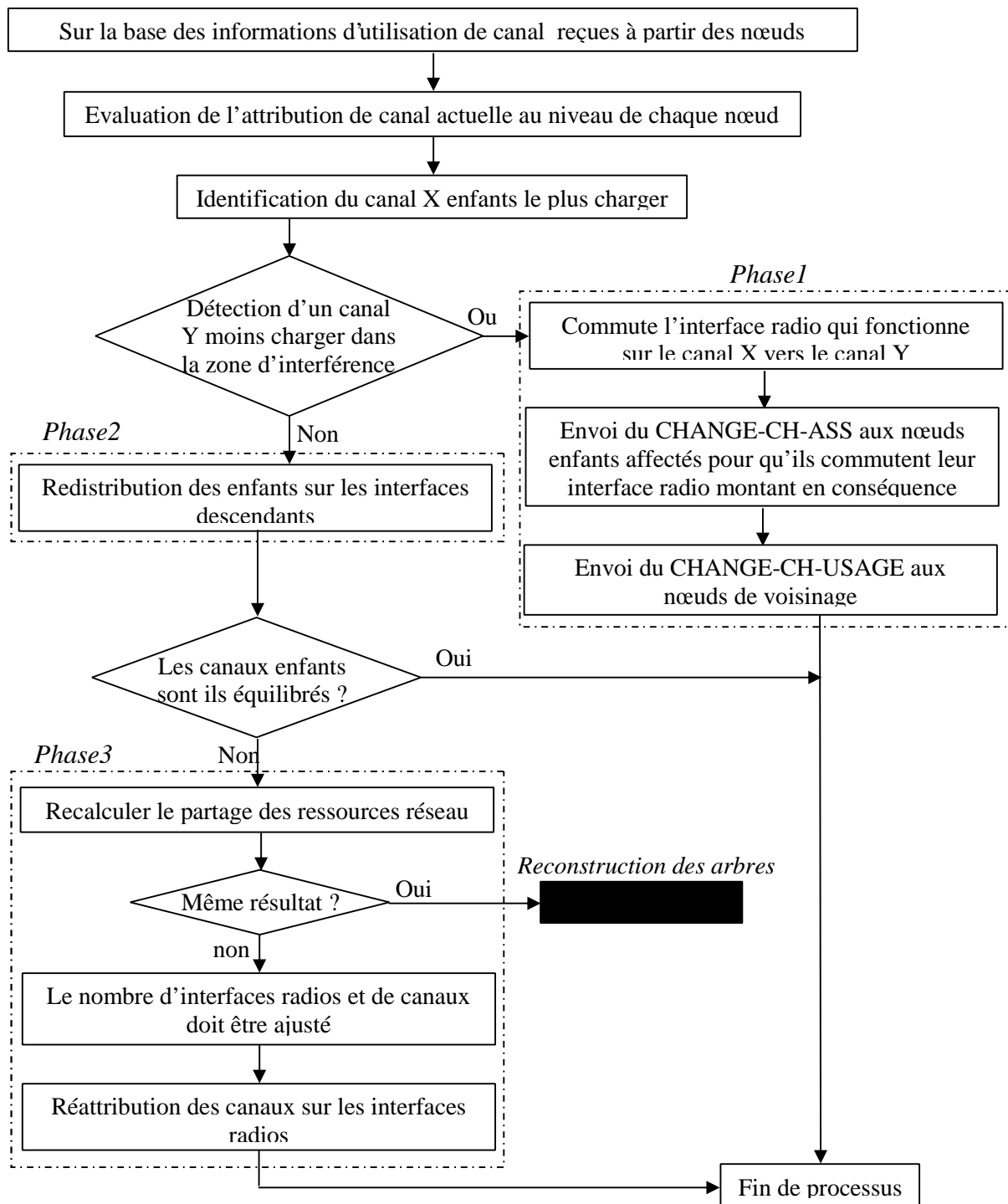


Figure 4.14 : Processus de réattribution de canal

Comme le montre le diagramme de la figure 4.14, le processus de réattribution de canal se compose de trois phases : '*Phase1: d'équilibrage de charge de canaux*'; '*Phase2: d'équilibrage de charge des interfaces enfants*' et ; '*Phase3: réinitialisation*'. Si après ces trois phases, les canaux restent encore déséquilibrés, Une reconstruction des arbres de routage aura donc lieu.

#### **4.4.3.2 Équilibrage de charge avec reconstruction de l'arbre de routage**

Le but principal dans le processus de construction des arbres de routage était bien l'équilibrage de charge. Pour garder cette condition toujours satisfaite, chaque nœud dans le réseau maintient une liste de parents potentiels qui sont ses voisins. Ceux-ci ne sont ni des nœuds descendants actuels ni des parents actuels. Il met ensuite à jour leur métrique de routage lorsqu'il reçoit (les messages TR-CONST reçu pour la construction de chaque arbre). Quand un nœud trouve un parent potentiel avec une meilleure métrique de routage, il tente de rejoindre le nouveau parent et se déconnecte de l'ancien parent. Alors aussi bien que le nouveau nœud parent que l'ancien nœud parent, ils informent leurs nœuds d'ancêtre de cet événement de changement de chemin et les nœuds d'ancêtre mettent à jour leurs entrées dans les différentes tables de routage relatives au nœud et ses descendants. Puisque les messages TR-CONST sont périodiquement diffusés, chaque nœud compare constamment les parents potentiels et choisit le meilleur pour s'y associer. Ce mécanisme permet au réseau de pouvoir continuellement se réarranger pour équilibrer les charges sur les nœuds.

Néanmoins, un problème peut survenir si tous les nœuds choisissent simultanément un même parent qui est moins chargé. Cette simultanéité de commutation peut surcharger ce parent. Ce problème est connu par le nom de 'route flapping'.

Nous avons trouvé deux solutions pour ce problème.

La première solution présentée dans [26] est inspirée des réseaux filaires. Etant donné que la passerelle est le seul nœud conscient de sa dernière charge de liaison, le message JOINT sera propagé jusqu'à la passerelle. La passerelle peut maintenant répondre par un ACCEPTE ou un REJET en arrière du nœud nouvellement joignant, basé sur la dernière capacité de liaison résiduelle de la passerelle. De plus, n'importe quel nœud intermédiaire peut aussi envoyer un message de REJET à de nouvelles demandes si sa capacité a diminué parce que d'autres nœuds ont commutés pour rejoindre son sous arbre. De plus, le message RT-ADD met aussi à jour l'utilisation de liaison actuelle sur chaque saut du chemin.

Dans [38], pour prévenir ce problème, la solution exige que chaque nœud planifie de commuter leur parent par un choix aléatoire d'une valeur de minuteur et la commutation se

fait seulement quand le minuteur expire et les charges de passerelle ne sont toujours pas équilibrées à ce point.

La première solution basée sur le calcul de capacité de chemin engendre plus d'overhead dans le réseau, par rapport à la deuxième solution, et ce à cause des messages de communication supplémentaires avec la passerelle. Parallèlement, la deuxième solution basée sur le choix aléatoire d'une valeur de minuteur peut causer des pertes de temps si le choix aléatoire donne une même valeur.

Après reconstruction de l'arbre, les nœuds affectés par ce processus doivent ré-exécuter le processus d'attribution de canal. Pour cela, la période de réattribution de canal doit être inférieure à la période de reconstruction de l'arbre de routage.

Le diagramme suivant résume le processus d'équilibrage de charge avec reconstruction de l'arbre de routage :

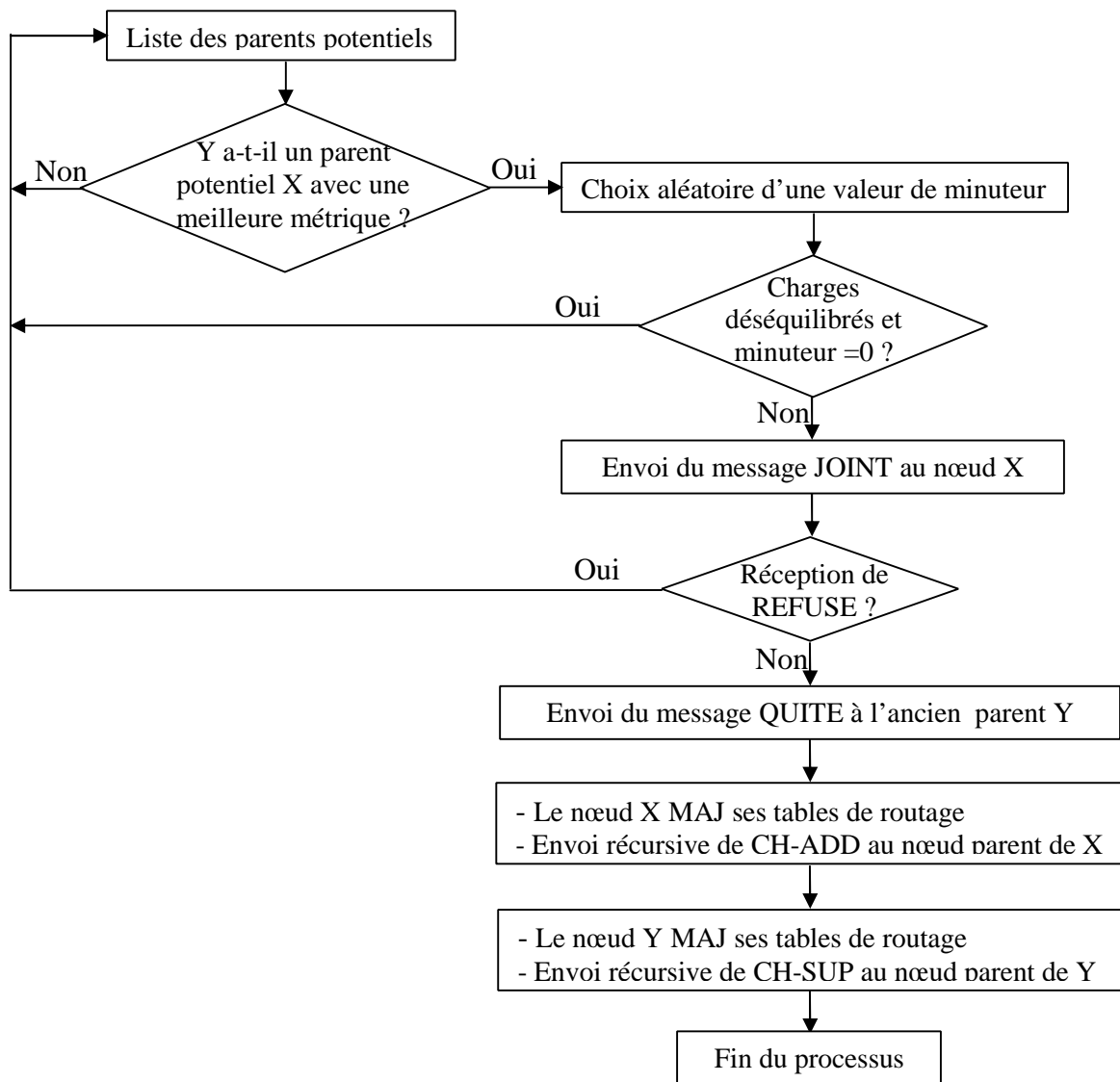


Figure 4.15 : Processus de reconstruction de l'arbre de routage

Les messages utilisés dans la reconstruction de l'arbre de routage sont décrits dans le tableau suivant.

**Tableau 4.7 : Table des messages échangés dans la reconstruction des arbres**

Message	Détaille
REFUSE	Réponse négative au message JOINT
QUITE	Envoyé a l'ancien nœud parents, après réception de message ACCEPT a partir du meilleur nouveau parent détecté.
CH-ADD	Envoyé au nœud parent après jointure d'un nouveau nœud enfants.
CH-SUP	Envoyé au nœud parent après le départ d'un nœud enfants.

#### 4.4.4 Tolérance aux pannes

Un WMN devrait être capable de dévier rapidement le trafic autour des nœuds en panne afin de maintenir une connectivité de réseau robuste en présence de panne.

##### 4.4.4.1 Processus de détection et de rétablissement de pannes

Pour exécuter un routage insensible aux défaillances, chaque nœud contrôle constamment les messages TR-CONST de ses parents pour déterminer s'ils sont inaccessibles à cause d'une panne de nœud ou d'une liaison. Chaque fois qu'un nœud, dit N, détecte que l'un de ses parents est inaccessible, il essaye rapidement de s'associer avec ses descendants ou avec un des parents de secours de N pour pallier à ce problème.

Si le nœud N peut s'associer avec succès avec ses descendants ou avec un de ses parents de secours, il ne doit pas informer ses descendants de la panne. Cependant, si aucun de ses parents de secours n'est accessible, N doit donc impliquer ses enfants dans le rétablissement de la panne. Particulièrement N enverra un message à ses enfants pour déclencher leur logique de rétablissement de panne, comme si la liaison entre N et ses enfants est morte. Après ce message, les enfants de N essayeront de connecter avec leurs parents de secours et N lui-même basculera à l'état silencieux pour attendre jusqu'à un nouveau candidat parental (très probablement un de ses anciens enfants) montre. Si nécessaire, la même procédure de rétablissement de la panne distribuée est appliquée récursivement sur la sortie de sous arbre entier sous N, jusqu'à découverte d'une liaison à d'autres arbres recouvrant ou la conclusion que le sous arbre de N est isolé du reste du réseau.

Le diagramme suivant représente le processus de détection et de rétablissement de panne.

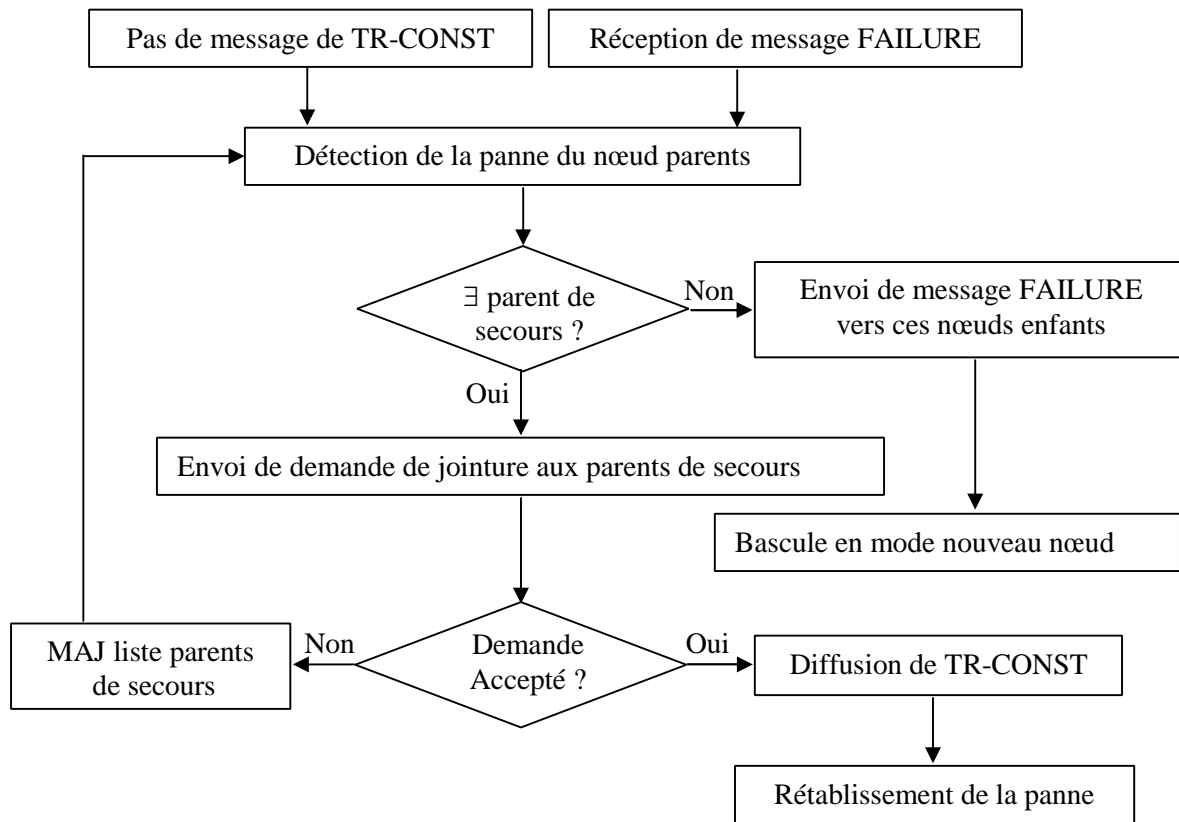
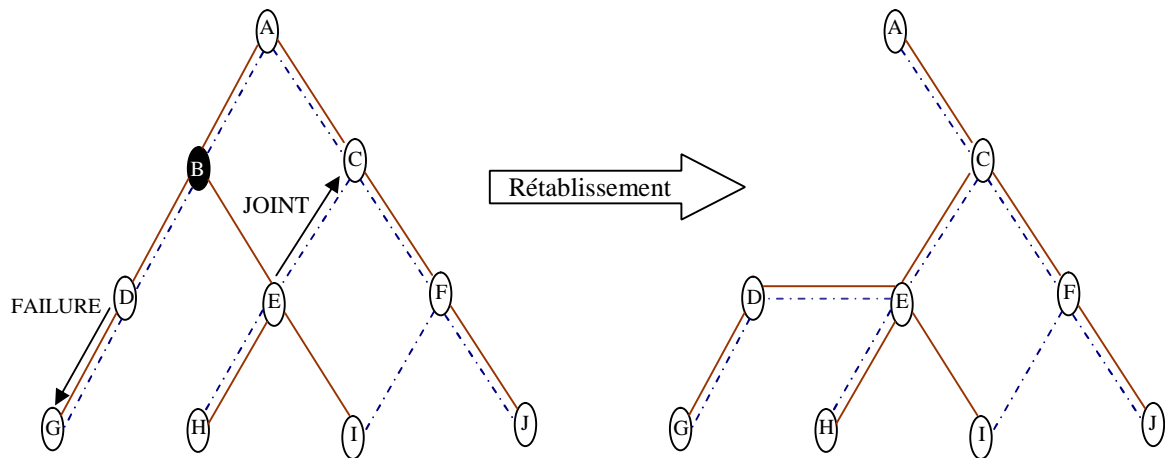


Figure 4.17 : Processus de détection et de rétablissement de pannes

#### 4.4.4.2 Exemple

Comme indiqué dans la figure 4.18 (a). Ici, quand le nœud B tombe en panne, son nœud enfant D n'a aucun parent de secours qui peut le reconnecter au réseau câblé. Il envoie donc le message FAILURE à son seul enfant G lui demandant d'exécuter leur rétablissement propre de la panne et se bascule en mode nouveau nœud. L'autre nœud enfant E par contre a comme seul parent de secours l'autre parent C de l'arbre BE. Pour réparer cette panne, le nœud enfant E envoie alors le message JOINT au nœud C. La figure 4.18 (b) montre la connectivité de réseau final après que le processus de rétablissement des liaisons est achevé.

Un rétablissement de la panne est finalement suivi par la définition du profil de trafic périodique habituel et des rajustements de canal pour rééquilibrer la charge du réseau à travers les différents canaux.



(a) Envoi des messages FAILURE, JOINT après la panne de B

(b) Nouveaux arbres après rétablissement

**Figure 4.18 : Exemple de rétablissement de pannes**

## 4.5 Conclusion

On s'attend à ce que l'accès sans fil à la large bande ait une architecture à trois niveaux, avec le backbone mesh sans fil pour expédier le trafic entre les réseaux d'accès (comme WLANs, ad hoc, cellulaires, etc.) et le backbone d'Internet. Dans ce chapitre, nous avons proposé un protocole de routage basé sur un multi arbres de routage et un algorithme d'attribution de canal avec différenciation de service comme une approche prometteuse pour le support de QoS sur le backbone maillé sans fil.

Dans le chapitre suivant, nous allons présenter les résultats de la simulation de notre proposition.

# 5

---

## Simulation et résultats

---

### 5.1. Introduction

La simulation permet de tester à moindre coût les nouveaux protocoles et d'anticiper les problèmes qui pourront se poser dans le futur afin d'implémenter la technologie la mieux adaptée aux besoins. NS [47] est un simulateur à événements discrets disponible gratuitement sur le site <http://www.isi.edu/nsnam/>. Il permet à l'utilisateur de définir un réseau et de simuler des communications entre les nœuds de ce réseau.

Dans ce présent chapitre, nous allons présenter le simulateur NS2, que nous allons utiliser pour évaluer les performances du protocole qu'on a présenté dans le chapitre 4. Nous allons, tout d'abord, éclaircir certaines fonctionnalités du simulateur NS2, telles que la prise en charge des réseaux sans fil et le support de multi-interfaces. Nous expliquons par la suite la simulation proprement dite. Celle-ci comprend plusieurs phases : implémentation, scénario de simulation et analyse des résultats.

### 5.2. Présentation du simulateur NS2

Le simulateur du réseau NS2 est un outil logiciel de simulation de réseaux informatiques. Il est principalement bâti avec les idées de la conception par objets, de réutilisation du code et de modularité.

NS2 est écrit en C++ et utilise le langage OTCL (Object Tools Command Language) dérivé de TCL. A travers OTCL, l'utilisateur décrit les conditions de la simulation : la topologie du réseau, les caractéristiques des liens physiques, les protocoles utilisés et les communications qui ont lieu. La simulation doit d'abord être saisie sous forme de fichier que NS va utiliser pour produire un fichier contenant les résultats.

NS2 contient les fonctionnalités nécessaires à l'étude des algorithmes de routage unicast ou multicast, des protocoles de transport, de réservation, des services intégrés et des protocoles d'application. De plus le simulateur possède déjà une palette de systèmes de transmission, d'ordonnanceurs et de politiques de gestion de files d'attente pour effectuer des études de contrôle de congestion. Pour une documentation générale de NS, on a référé à [47] [49].

Un modèle de réseau sous NS est constitué de :

- Nœuds de réseau : endroits où est généré le trafic, ou nœuds de routage ;
- Liens de communication entre les nœuds du réseau.
- Agents de communication, représentant les protocoles de niveau transport (TCP, UDP) ; ces agents sont attachés aux nœuds et connectés l'un à l'autre, ce qui représente un échange de données (connexion TCP, flux UDP).
- Applications qui génèrent le trafic de données selon certaines lois (CBR, VBR), et se servent des agents de transport.

### 5.3. Support du Multi-Interface, Multi-Canaux dans NS2

La croissance rapide de la technologie IEEE 802.11 a atténué la baisse sensible des prix des produits correspondants et, par conséquent, leur présence est devenue de plus en plus fréquente. Ce qui a suscité l'intérêt de la communauté du simulateur de réseau NS. Beaucoup de chercheurs sont d'ailleurs prêts à étendre leurs modèles de simulation pour intégrer des interfaces multiples

La figure 5.1 montre l'architecture d'origine d'un nœud mobile, qui consiste, après l' 'agent de routage', d'une chaîne de modules, qui émule la pile protocolaire. Les différentes entités que tout hôte aurait dans la vie réelle sont: 'la couche liaison', 'la couche mac', 'le module arp', 'la file d'attente d'interface' et 'l'interface réseau'. Ces modules sont tous connectés au même canal partagé sans fil.

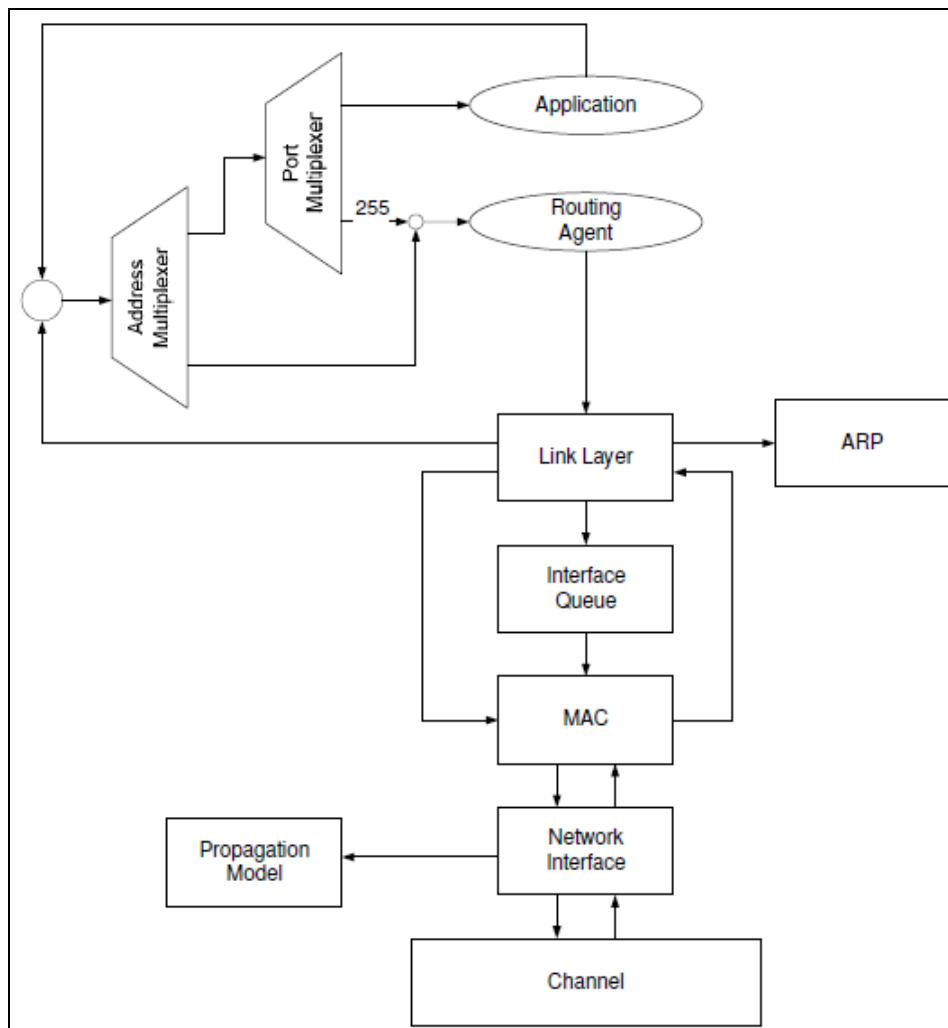


Figure 5.1 : Architecture d'un nœud mobile

La figure 5.2 [50], présente l'architecture de haut niveau du nœud mobile 'modifié'. Comme on le voit, chaque nœud a autant d'exemplaires de la chaîne d'entités que d'interfaces dont il dispose.

Pour le trafic entrant, il n'y a pas beaucoup de différences par rapport à l'opération originale du simulateur. Les paquets entrants arrivent par le canal auquel ils correspondent et passent à travers les différentes entités dans l'ordre croissant puisque le dernier module de chaque interface, c'est-à-dire la «couche liaison», est connecté au même point commun ('multiplexeur d'adresse') et tous les paquets sont traités par l'agent approprié (soit de routage ou d'application) indépendamment de l'interface d'origine ou les paquets sont arrivés.

D'autre part, pour le trafic sortant, il convient de souligner que l'intelligence de choisir l'interface appropriée doit être au sein de l'agent de routage, comme on le voit dans la figure précédente, c'est le point dans lequel la décision doit être prise.

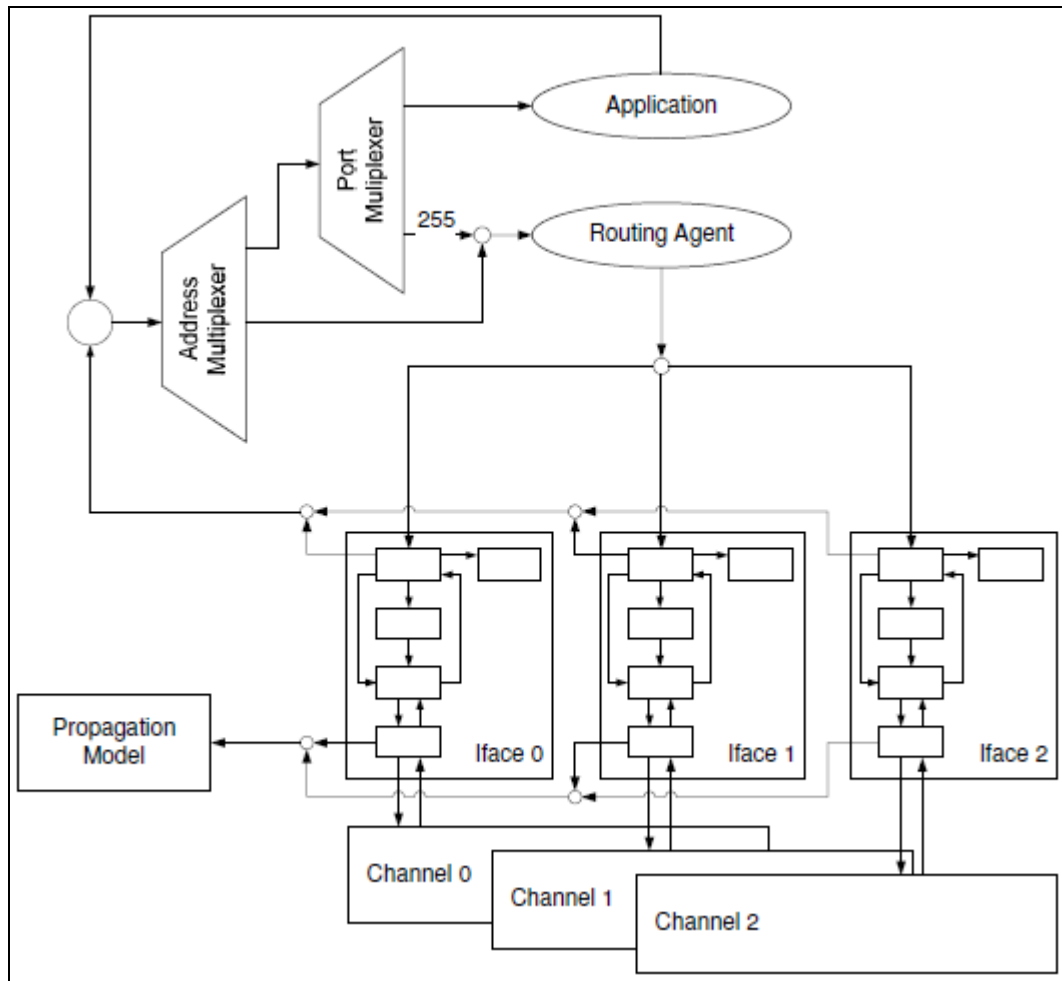


Figure 5.2 : Architecture modifier d'un nœud mobile, avec support de multi interface

Pour plus de détails sur les changements nécessaires dans le code TCL et C++ pour l'implémentation du support du multi interfaces dans NS, il faut se rapporter aux détails présentés en annexe 2 et les références [50] [51] [52].

#### 5.4. Implémentation du protocole CA-DiffServ proposé

Le protocole présenté dans le chapitre précédent a été implémenté en utilisant le simulateur de réseau NS. Ce développement représente environ 4500 lignes de code en C++ et en TCL, NS intègre, dans sa version actuelle, quelque protocole de routage pour réseau adhoc comme AODV, TORA, DSR et DSDV, et supporte le modèle de différenciation de service pour les réseaux filaires seulement. Le choix de NS pour la simulation de CA-DiffServ vient de l'importance de la communauté des utilisateurs, assurant que ce simulateur propose une modélisation relativement réaliste des phénomènes physiques et une implémentation fiable des protocoles inclus dans ces distributions.

Ce développement représente une part importante du travail effectué durant ce projet car, si NS est un outil largement utilisé dans la communauté scientifique, il souffre d'une documentation peu claire surtout dans le cas des réseaux maillés multi-radio multi-canaux. Il est souvent nécessaire de se reporter directement au code source du simulateur pour résoudre un problème particulier.

Dans ce qui suit, nous allons décrire le processus d'implémentation de notre protocole CA-DiffServ. Ceci se fera sous la forme standard des algorithmes distribués qui sont conçus suivant un ensemble d'évènements discrets matérialisés chacun par une procédure événementielle.

#### **5.4.1. Modélisation du protocole sous forme événementielle**

La forme événementielle du protocole nous a conduite à l'élaboration d'un ensemble de procédures. Parmi cet ensemble, nous retrouvons trois procédures principales : «advertisse», «measurement» et «load\_balancing». Ces procédures sont exécutées lors de démarrage de chaque nœud et elles sont relancées périodiquement avec des délais différents.

- La première procédure, 'Advertisse' sert à diffuser le message 'ST\_CONST' pour la construction des arbres de routage.
- La deuxième procédure 'measurement' consiste à mesurer la charge des nœuds et des canaux dans le réseau et déclenche une éventuelle réattribution de canaux.
- La troisième procédure 'load\_balancing' consiste à équilibrer la charge du réseau commençant par les interfaces enfants du nœud jusqu'à la reconstruction de l'arbre si besoin.

Ces procédures invoquent d'autres procédures correspondantes aux évènements suivants:

- Demande de jointure/quitte de l'arbre
- Réponse à la demande de jointure
- Ajout/suppression d'un enfant
- Mise à jour de l'utilisation des canaux

Il existe, également, d'autres procédures secondaires que nous n'avons pas citées, mais qui sont appelées à partir des procédures que nous avons déjà évoquées. Pour plus de détail, il faut se rapporter à l'annexe 3.

### 5.4.2. Réalisation des testes et interprétation des résultats

NS2 écrit les résultats de ses simulations dans un fichier texte où chaque ligne correspond à un évènement qui s'est produit à un niveau ou à un autre de la pile protocolaire. Il est possible de configurer NS2 de telle sorte qu'il ne garde une trace que de certains types d'évènements (par exemple tout ce qui concerne le routage, mais pas ce qui concerne la couche MAC). Ceci est en particulier utilisé pour accélérer la simulation et réduire la taille du fichier.

Ce fichier texte peut porter en lui même énormément d'informations. Mais pour extraire et représenter de manière synthétique ces informations, il faut souvent appliquer de nombreux traitements à ce fichier.

L'analyse des fichiers de trace dans nos simulations est réalisée en utilisant le langage puissant AWK [53] pour extraire les champs nécessaires au calcul des différents paramètres du protocole. Enfin les graphes sont obtenus en utilisant GNUPLOT [54]. Une description détaillée sur le fichier de trace et le langage AWK est donnée dans l'annexe 4.

## 5.5. Evaluation de performance

### 5.5.1. Métrique de construction de l'arbre de routage

Comme déjà mentionné dans le chapitre 4, nous avons amélioré la métrique de construction de l'arbre de routage proposé dans [26] par l'ajout de deux paramètres essentiels pour l'équilibrage de charge, qui sont le nombre de voisins d'un nœud et le nombre d'interfaces radio dont dispose un nœud. Afin de montrer l'efficacité de notre protocole dans l'obtention des arbres de routage plus équilibrés, nous prenons un exemple de topologie grille de réseau constitué de 9 nœuds, comme montre la figure suivante.

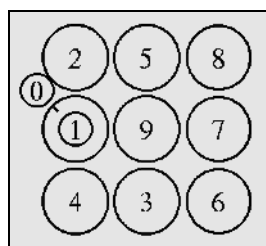


Figure 5.3 : Topologie grille d'un réseau constitué de 9 nœuds

Nous désignons le nœud 1 comme passerelle. Les figures 4A, 4.B montre les arbres de routages construits par l'algorithme 'Hyacinth' [48] et le notre respectivement.

Le nœud 5 reçoit le message ST\_CONST depuis le nœud 9 et le nœud 2, en exécutant l'algorithme 'Hyacinth', il choisit le nœud 9. Par contre, avec notre algorithme, il préfère le

nœud 2 comme parent, car le nœud 9 a plus de voisins (4voisins) que le nœud 2 (que 2 voisins). Même remarque pour le nœud 3.

Le lien qui relie la passerelle avec le nœud 9 dans la figure 5.4 (A) va recevoir plus de trafic par rapport au lien qui relie la passerelle avec le nœud 2 et 4, donc, on aura des liens déséquilibrés, chose qui est bien évitée dans la figure 5.4 (B) où les trois liens sont bien équilibrés.

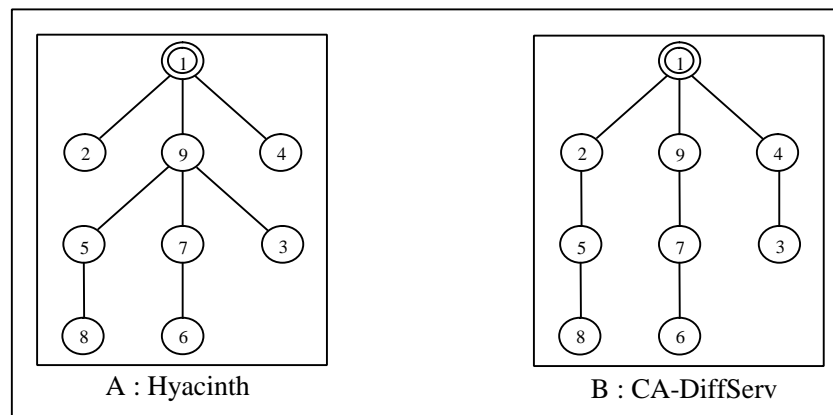


Figure 5.4 : Arbre de routage construit par les deux algorithmes

## 5.5.2. Métrique de performance

### 5.5.2.1. Latence

La latence ou délai de réponse caractérise le retard entre l'émission et la réception d'un paquet. En général, un protocole de routage est meilleur si la latence est faible.

La latence est l'un des paramètres les plus importants lors de l'analyse des performances dans les protocoles de routage conscients de QoS. Cette mesure évalue la capacité du protocole à utiliser les ressources du réseau efficacement. La latence moyenne est la moyenne sur l'ensemble des délais d'acheminement de bout en bout des paquets transmis avec succès.

### 5.5.2.2. L'équité

Le problème majeur de la différenciation de service sur Internet est le problème de famine du trafic best effort qui est moins prioritaire par rapport au trafic QoS. Les algorithmes d'ordonnancement proposés dans le modèle de DiffServ privilégient toujours le trafic QoS, ce qui pénalise le trafic BE. Un protocole de routage est meilleur s'il répond aux exigences des applications QoS tout en satisfaisant les applications Best Effort.

### 5.5.2.3. Frais généraux de routage

Les frais généraux de routage se mesurent soit par le nombre de paquets de contrôle qui est le nombre total de paquets de routage transmis au cours de la simulation soit par la charge de routage normalisée.

Les paquets de routage servent notamment à la construction des arbres de routage, l'équilibrage de charge des canaux ..., la mesure des envois et des transmissions. Les frais généraux de routage sont une valeur importante pour la comparaison des protocoles de routage car ils mesurent l'évolutivité et l'efficacité du protocole. Les protocoles qui envoient un grand nombre de paquets de routage peuvent également augmenter la probabilité de collisions de paquets et peuvent retarder les paquets de données dans les files d'interface réseau de transmission.

La charge de routage normalisée est le rapport entre nombre de paquets de routage envoyés et le nombre de paquets de données livré à destination. En général, un protocole de routage est meilleur si la charge de routage normalisé est faible.

$$CRN = \frac{\text{Nombre de paquet de contrôle envoyé}}{\text{Nombre de paquet de donnée reçu}}$$

### 5.5.3. Scénario de simulation

Dans nos simulations et nos comparaisons, et par mesure de simplicité, nous avons implémenté, dans un premier temps, l'attribution de canal hybride où l'interface zéro de chaque nœud utilise le canal zéro pour l'envoi et la réception des messages de contrôle.

Nous avons effectué une série de tests et de comparaisons pour notre protocole baptisé 'CA-DiffServ' avec le protocole Hyacinth comme exemple de protocole multi-canaux multi-interfaces et le protocole DSDV comme exemple de protocole mono-canal mono-interface pour générer les graphes de latence, d'équité et ceux d'overhead.

Voici les paramètres par défaut pour tous les scénarios de simulation. Chaque nœud est équipé de 5 cartes réseau, et le nombre de canaux physiques est de 6. Le rapport entre la zone d'interférence et la portée de communication est fixé à 2. La période d'équilibrage de charge des canaux  $T_c$  d'un nœud est fixé à une minute, tandis que la fréquence de mesure de l'utilisation des canaux et de construction des arbres, est fixé à 30 secondes ( $T_a$ ).

Le modèle de trafic CBR a été utilisé. Nous avons utilisé le protocole IEEE 802.11 DCF de la couche MAC, avec un débit de données maximal de 1 Mb/s. La taille de la file d'attente totale (dans toutes les priorités) a été fixée à 50 paquets.

Nous avons utilisé plusieurs topologies grille comme celle montrée dans la figure 3, composées de 9, 16, 25, 36 et 49 nœuds. Les nœuds adjacents sont dans la portée de transmission, tandis que les nœuds opposés diagonalement ne sont pas dans la portée de

transmission. Nous avons choisi le nœud passerelle celui qui se trouve dans le coin bas gauche pour avoir des chemins avec plus de nombre de sauts dans le calcul de la latence.

#### 5.5.4. Analyse des résultats

##### 5.5.4.1. Latence

Pour mesurer la latence des différents protocoles, nous avons inséré deux trafics : l'un de la classe QoS et l'autre de la classe BE dans tous les nœuds du réseau. Nous avons ensuite calculé la latence moyenne pour chaque trafic, dans cinq topologies avec des tailles différentes. Pour rendre le protocole 'Hyacinth' conscient de la QoS nous avons lui associe comme politique d'ordonnancement 'CMUPriQueue', cette politique de gestion de fil d'attente définie dans NS privilège les flux QoS par rapport aux flux BE. La figure 5.5 montre les graphes obtenus pour chaque flux dans chaque protocole.

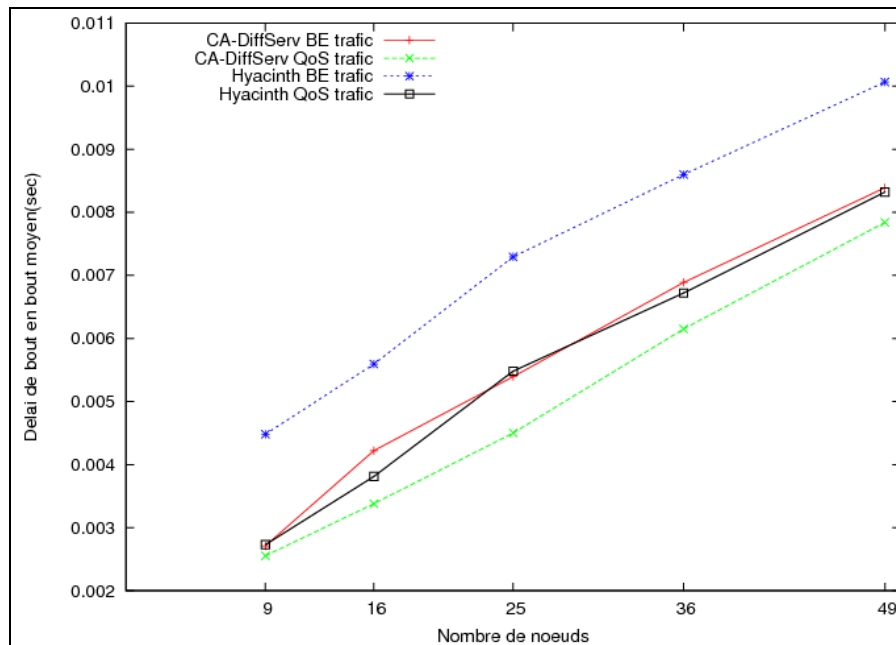


Figure 5.5 : Latence moyenne (CA-DiffServ Vs. Hyacinth)

Nous remarquons que notre protocole donne les meilleurs résultats, car il maintient deux routes différentes pour les deux classes de trafic. Tandis que le protocole 'hyacinth' travaille sur un même chemin pour acheminer les trafic des deux classes. . Si on compare la latence pour le trafic QoS, nous trouvons que notre protocole minimise mieux la latence, et ce grâce aux arbres de routage construits qui sont plus équilibrés que ceux du protocole 'hyacinth' comme expliqué dans le point 5.5.1.

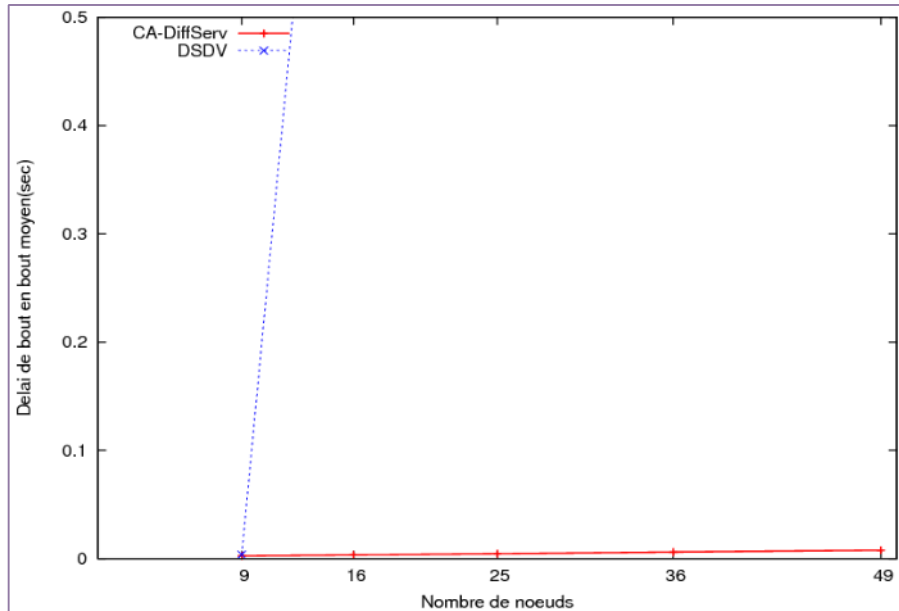


Figure 5.6 : Latence moyenne (CA-DiffServ Vs. DSDV)

Si on compare notre protocole avec un protocole mono-interface mono-canal, nous trouverons qu'il y a une grande amélioration qui est due à l'utilisation de plusieurs canaux et l'amélioration de la bande passante total du réseau.

#### 5.5.4.2. Équité

La figure 5.7 montre que l'équité entre les deux classes de trafic est bien satisfaite par notre protocole car les trafics des deux classes se lancent en parallèle. Par contre avec le protocole 'Hyacinth', le trafic BE se lance après le trafic QoS tout en respectons l'algorithme d'ordonnancement.

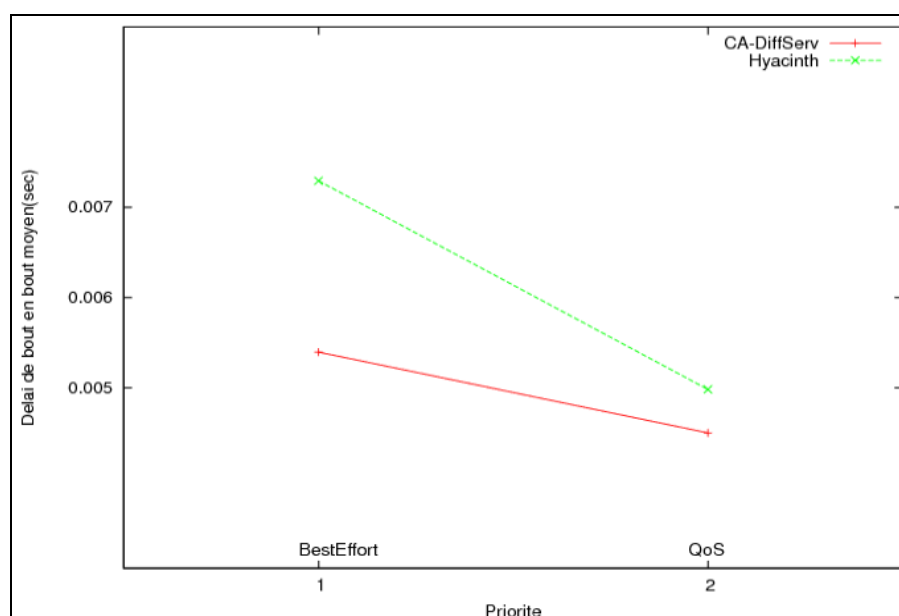


Figure 5.7 : Équité en latence (CA-DiffServ Vs. Hyacinth)

### 5.5.4.3. Influence de nombre de canaux sur la latence

Le nombre de canaux non chevauchés est de 3 dans la norme 802.11b/g et 12 pour 802.11a. La figure 5.8 montre l'effet de variation du nombre de canaux sur la latence moyenne. Ces résultats sont obtenus dans une topologie de réseau grille avec 16 nœuds. Les deux types de trafic sont générés aléatoirement dans 8 nœuds qui ont une distance de 1, 2, 3 et 4 sauts de la passerelle.

La latence diminue de façon monotone avec le nombre de canaux car plus le nombre de canaux augmente, moins sera le nombre de nœud qui partage le même canal est donc, l'interférence diminue et il est constaté moins de collisions.

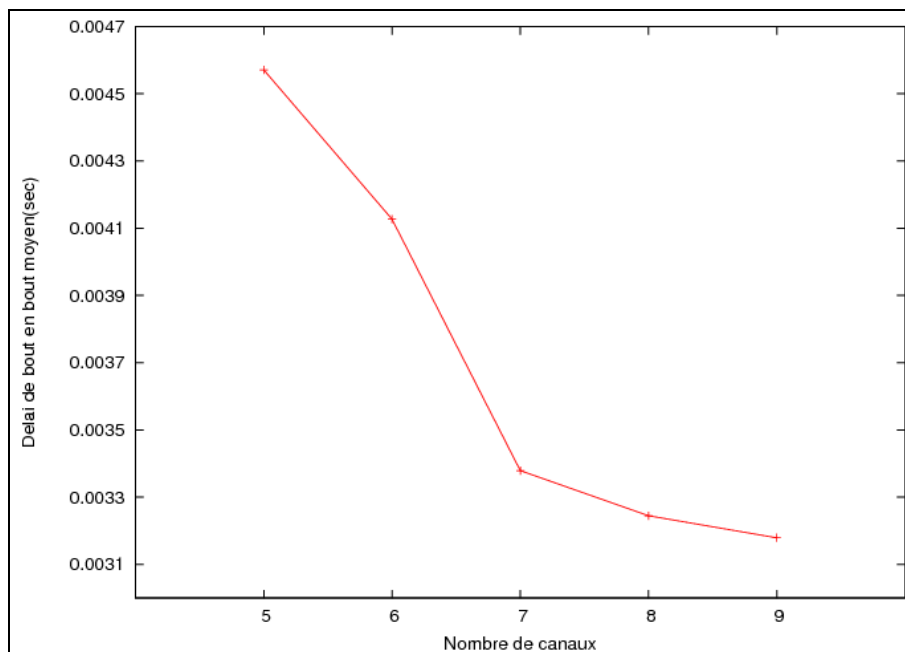
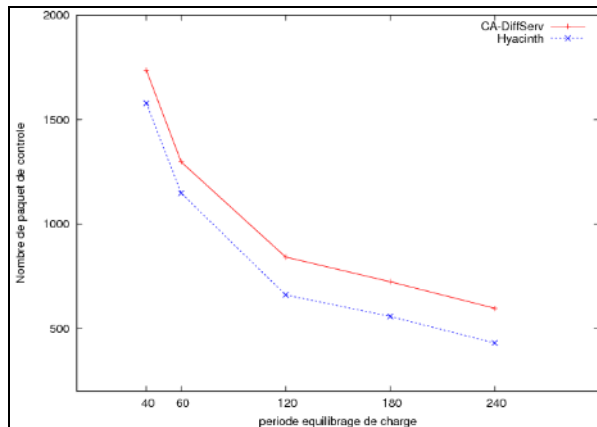


Figure 5.8 : Latence en fonction de nombre de canaux

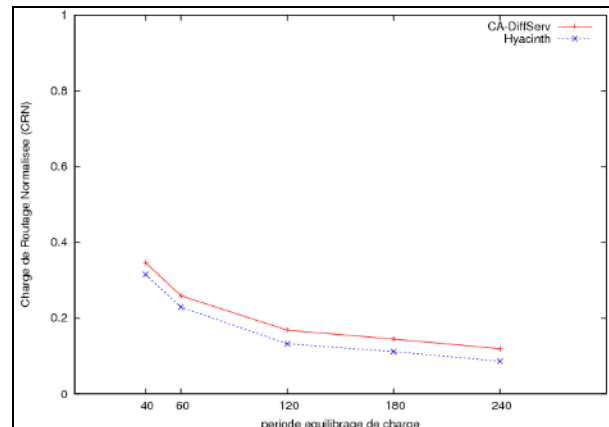
### 5.5.4.4. Overhead du protocole

Dans notre protocole, les nœuds du réseau suivent en permanence les charges sur leurs interfaces, échangent des informations sur l'utilisation des canaux toutes les 30 secondes et font des décisions d'équilibrage de charge toutes les 60 secondes. La bande passante du réseau devrait s'améliorer à mesure que les nœuds modifient leurs attributions de canaux et de routage.

Les figures 5.9 et 5.10 montrent le nombre de messages de contrôle envoyés et la charge de routage normalisé en fonction de la période d'équilibrage de charge. Pour des périodes d'équilibrage de charge élevée, le réseau génère moins de messages de contrôle ainsi la charge de routage normalisé diminue. Différentes périodes d'équilibrage de charge engendrent différents frais de communication.



**Figure 5.9 : Overhead en terme de nombre de paquets de contrôle**



**Figure 5.10 : Overhead en terme de charge de routage normalisé**

Nous remarquons que le protocole 'Hyacinth' engendre moins de paquets de contrôle que notre protocole mais la différence reste marginale. Nous pouvons justifier cette différence par le fait que notre protocole gère l'équilibrage de charge dans deux arbres de routage pour les deux types de trafic. Par contre, le protocole 'Hyacinth' s'intéresse à un seul arbre de routage et ne différencie pas entre les deux classes de trafic.

## 5.6. Conclusion

Les simulations présentées dans ce chapitre ont permis de valider le fonctionnement de notre protocole. Elles indiquent, en particulier, comment le protocole réagit face à la densité du réseau et à la charge du trafic circulant dans le réseau. Et ceci en termes de l'équilibrage de charge des routes et des canaux, délai de réponse ainsi que le l'évitement du problème de famine. La comparaison avec le protocole 'Hyacinth', nous a permis de dégager le comportement du protocole face aux connexions de QoS et celles de Best effort suite à la construction des arbres de routage multiple.

Toutefois, les performances réelles de ce protocole ne peuvent pas être déduites à partir de simples simulations. Il est en effet difficile, en simulation, de concevoir les scénarios pertinents conduisant à l'évaluation du comportement du protocole.

---

# Conclusion générale

---

WMNs sont prévus comme une solution clef de la demande croissante sur la gestion de multimédia dans le réseau et la fourniture de l'accès à haut débit. Le coût réduit et la facilité d'installation du WMNs comparés à leur contrepartie filaire poussent l'industrie et le monde universitaire à prêter plus d'attention à cette technologie prometteuse.

Le problème d'attribution de canal dans les réseaux maillés sans fil est un problème difficile, compte tenu de nombreuses contraintes imposées par le médium radio et par la charge de trafic des nœuds dans un tel réseau. Si les travaux menés jusqu'ici tentent de résoudre les problèmes liés à l'amélioration de performance de réseau en présence d'un seul type de trafic, aucun des protocoles, proposés jusqu'ici, ne prend en compte le problème de famine entre les trafics appartenant à différentes classes de priorité.

C'est dans l'optique d'étudier ce phénomène et d'apporter une solution à ce problème que nous avons conçu le protocole CA-DiffServ. Ce protocole apporte aux nœuds du réseau une connaissance de leur voisinage suffisante pour leur permettre de connaître les canaux les moins utilisés pour les affecter à leurs interfaces.

Le travail effectué a permis d'une part d'évaluer l'impact de l'utilisation de plusieurs interfaces réseaux et plusieurs canaux, et de montrer qu'elles pouvaient fortement augmenter le débit du réseau et d'autre part de proposer un début de solution au problème d'attribution de canal sous la coexistence de différents classes de trafic. L'implémentation actuelle du protocole CA-DisffServ sous le simulateur de réseaux NS a permis une première évaluation.

Les tests et les comparaisons ont montré différents points :

- La métrique de construction des arbres de routage introduite dans CA-DiffServ permet d'équilibrer la charge dans le réseau et donc maintenir des délais faibles pour les trafics;
- L'utilisation de plusieurs canaux améliore considérablement le débit du réseau, diminue l'interférence et par conséquence réduit énormément la latence de trafic.

- La construction de plusieurs arbres de routage, un arbre pour chaque classe partage les ressources du réseau de façon efficace et équitable entre les flux concurrents.

En conclusion, le protocole CA-DiffServ s'est avéré efficace pour empêcher le problème de famine entre les classes de trafic.

Cependant, il ne s'agit là que d'une première étape vers un protocole d'attribution de canal avec différenciation de service réellement efficace. En effet, si la version actuelle de CA-DiffServ, telle qu'elle est décrite dans ce document, permet de résoudre ces problèmes dans bon nombre de cas, les simulations effectuées ont permis de révéler plusieurs points importants qui devront être améliorés.

D'une part, en le comparant avec le protocole 'Hyacinth', notre protocole engendre plus de paquets de contrôle, et nécessite donc plus de frais de communication et c'est le cas pour tous les protocoles multi-chemin. Trouver une solution qui minimise ces frais s'avère donc nécessaire pour pouvoir bénéficier de tous les ressources du réseaux.

D'autre part, dans notre protocole, par mesure de simplification, nous n'avons considéré que deux classes de trafic (QoS et Best Effort) et son efficacité n'est pas encore testée pour un nombre plus important de classes. En effet, dans le modèle DiffServ proposé par l'IETF, trois classes de trafic sont définies avec des sous classes modélisant ainsi le trafic réel dans Internet. Une solution à ce problème pourrait consister à garder un arbre principal pour le trafic de la classe d'excellence et un autre arbre secondaire pour la somme de trafics des autres classes et proposer un algorithme pour l'ordonnement de ces trafics.

Enfin, de nombreuses fonctionnalités ou optimisations peuvent être apportées à ce protocole.

- L'implémentation de la partie tolérance aux pannes qui consiste à la reconstruction des arbres dans le cas ou un nœud chargé de transmettre des paquets disparaît ou un lien devient défaillant.
- L'implémentation de la version purement dynamique proposée dans ce travail pour bénéficier de toutes les ressources réseaux.
- Une autre amélioration consisterait à utiliser un protocole MAC conscient de la QoS comme IEEE 802.11e pour avoir une solution de différenciation de service complète.
- Valider le protocole dans un environnement réaliste, sans les problèmes de simplification inhérente à toute modélisation. Les interférences radio sont notamment légèrement sous-estimées dans les modèles de simulation.

Les résultats obtenus dans ce travail sont prometteurs et nous encourageant à continuer dans cette direction.

---

# Références bibliographiques

---

- [1] -Layer et Gestion de la Mobilité dans les Réseaux Maillés Sans fil”, Thèse de Doctorat de l’Université Paris VI Pierre et Marie Curie, 2006.
- [2] Computer Networks, vol. 47, pp. 445-487, 2005
- [3] Miguel Elias M. Campista, Pedro Miguel Esposito, Igor M. Moraes, Luis Henrique M. K. Costa, and Otto Carlos M. B. Duarte, Diego G. Passos, Célio Vinicius N. and  
for Wireless Mesh Networks” IEEE Network, Volume 22, pp 6-12, Jan/Feb 2008
- [4] D. Passos, D. V. Teixeira, D. C. Muchaluat-Saade, L. C. S. Magalhaes, and C. V. N. de  
Information and Telecommunications Technologies Symposium (I2TS), Dec. 2006.
- [5] -radio, multi-hop wireless mesh networks,” in ACM International Conference on Mobile Computing and Networking (MobiCom), Sept. 2004,
- [6] IEEE Workshop on Wireless Mesh Networks (WiMesh), Sept. 2005.
- [7] -aware routing metrics for time-varying wireless mesh networks,” IEEE Journal on Selected Areas in Communications, vol. 24, no. 11, pp. 1984-1994, Nov. 2006.
- [8] A. P. Subra  
multi-radio wireless mesh networks,” in IEEE Workshop on Wireless Mesh Networks (WiMesh), Sept. 2006

- [9] Rainer Baumann and Vincent Lenders and Simon Heimlicher and Martin May, Proceedings of the IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2007.
- [10] Claude Chaudet, Isabelle Guérin Lassous "Routage QoS et réseaux ad-hoc : de l'état de lien à l'état de nœud" Rapport de recherche n° 4700, INRIA, France, 2003.
- [11] J. Jun et M. Sichitiu, "MRP: Wireless Mesh Networks Routing Protocol" Center for Advanced Computing and Communications (CACC), Raleigh, NC 27695-7911, 2006.
- [12] A. Tamer, "QoS Routing in Wireless Mesh Networks" thèse Master de Science Appliquée présenté à l'Université de Waterloo département Électronique et Informatique, Canada 2008.
- [13] A. Abdrabou and W. Zhuang, "A position-based qos routing scheme for uwb mobile ad hoc networks". IEEE Journal on Selected Areas in Communications, vol. 24, pp. 850-856, 2006.
- [14] F. Harivelo et P. Anelli, "Différenciation de services sur WLAN" IREMIA, Université de La Réunion, France 2005.
- [15] Hai Jiang, Weihua Zhuang, Xuemin (Sherman) Shen, Atef Abdrabou, et Ping Wang "Differentiated Services for Wireless Mesh Backbone" , Centre for Wireless Communications (CWC) Department of Electrical and Computer Engineering University of Waterloo, Waterloo, Ontario, Canada N2L 3G1, 2006.
- [16] Vinod Kone, Sudipto Das, Ben Y. Zhao and Haitao Zheng, "QUORUM – Quality Of service RoUting in wireless Mesh networks" IEEE/ICST International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine), University of California, 2007.
- [17] J. Tang , G. Xue and W. Zhang, "Interference-aware topology control and QoS routing in multi-channel wireless mesh networks", Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing, Urbana-Champaign, IL, USA, May 25-27, 2005.
- [18] R. Leung, J. Liu, E. Poon, C. Chan et B. Li. "DSR: A QoS-aware Multi-path Dynamic Source Routing Protocol for Wireless Ad-hoc Networks," in Proceedings of

- the 26th IEEE Annual Conference on Local Computer Networks (LCN 2001), pp. 132-141, Tampa, Florida, Nov 15-16, 2001.
- [19] S. Chen et K. Nahrstedt, "Distributed quality-of-service routing in ad hoc networks". IEEE Journal on Selected Areas in Communications", vol. 17, pp. 1488-1505, 1999.
- [20] networks", IEEE Communi. Surv. and Tutor., vol. 9, no. 2, pp. 50–70, 2007.
- [21] i-Radio Multi-Channel Wireless Ad Hoc Networks" 2007. <http://www.dei.unipd.it/~merlo/papers/mc4.pdf>
- [22] Nowsher Ali A Survey on Integrated Channel-assignment and Power-control in Wireless Ad Hoc Networks" Course: 60-510 School of Computer Science University of Windsor 2007.
- [23] Channel Multi-Interface Wireless networks" Technical Report, University of Illinois at Urbana-Champaign 2005.
- [24] H. Skalli, S. Ghosh, S.K. Das, L. Lenzini, M. Conti, for Multi-radio Wireless Mesh Networks: Issues and Solutions" Communications Magazine, IEEE V. 45, page(s): 86-95, Nov 2007
- [25] A. Raniwala, K. Gopalan, T. Chiueh. Centralized Algorithms for Multichannel Wireless Mesh Networks". ACM Mobile Computing and Communications Review (MC2R), April 2004.
- [26] A. Raniwala, T. Chiueh. Architecture and Algorithms for an IEEE 802.11-based Multi-channel Wireless Mesh Network". In Proceedings of IEEE Infocom, 2005.
- [27] Xin Wang , J.J. Garcia-Luna- and Scheduling for Wireless Mesh Networks" Department of Computer Engineering, University of California, [Computer Communications V31](#), Pages 1436-1446, 2008
- [28] K. Ramachandran, K. Almeroth, E. Belding-Aware Channel Assignment in Multi-Radio Wireless Mesh Networks". In Dept. of Computer Science Technical Report, University of California, Santa Barbara, Santa Barbara, CA, July 2004.

- [29] Channel Assignment in Multi-Channel Multi-Radio Wireless Mesh Networks – Extended Abstract” NEC Laboratories China 2007
- [30] -based Channel Assignment Scheme for Wireless Mesh Networks” Mobile Adhoc and Sensor Systems (MASS 2007). IEEE International Conference on page(s): 1-6, Oct 2007
- [31] Sok-Hyong Kim and Young- Assignment for Multi-Channel Wireless Mesh Networks” 2008
- [32] -layer Protocols for Multi-Channel Multi-Interface Ad Hoc Wireless Networks”, SIGMOBILE Mobile Computing and Communications Review 10(1), 31-43. 2006
- [33] assignment for multi-radio Wireless Mesh Networks” Technical report, 2006. [http://www.imtlucca.it/\\_documents/publications/publication54-9369-Technical\\_report\\_2006.pdf](http://www.imtlucca.it/_documents/publications/publication54-9369-Technical_report_2006.pdf)
- [34] K. Ramachandran, M. Buddhikot, G. Chandranmenon, S. Miller, E. Belding-Royer, Networks”. In IEEE WiMesh2005, San Jose, CA, Sept 2005
- [35] ch for Utilizing Multiple Channels in Multi-Radio Wireless Mesh Networks,” *Proc. Broadnets*, pp. 381–90. Oct 2005
- [36] telecommunication systems: A comprehensive survey,” IEEE Personal communications, pp. 10-31, 1996.
- [37] Ekram Hossain · Kin Leung Wireless Mesh Networks Architectures and Protocols” Springer Science+Business Media, LLC 2008
- [38] Switching on IEEE 802.11-Based Wireless Mesh Networks” Stony Brook University WICON’08, Nov. 2008

- [39] Karam, M.J. Tobagi, F.A  
Comput. Syst. Lab., Stanford Univ., CA; Global Telecommunications Conference, GLOBECOM '00. IEEE page(s): 548-554 vol.1, 2000
- [40] Protocol (AMSTP) for Optical Ethernet Backbones” Icn, page 744-751, 29th Annual IEEE International Conference on Local Computer Networks (LCN'04), 2004
- [41] Yujin Lim Heeyeol Yu Das, S. Lee, S.S. Gerla, M. -aware Multiple Spanning Tree Mechanism over a Bridged LAN Environment” Dept. of Comput. Sci., UCLA, Los Angeles, CA, USA; Global Telecommunications Conference, GLOBECOM '03. IEEE, page(s): 3068- 3072 vol.6 Dec. 2003
- [42] -radio multi-channel wireless mesh networks”. In *ACM SIGMOBILE*, 2005.
- [43] Pai- Load-Balancing Routing for Wireless Access Networks” Proceeding IEEE Infocom, pp. 986-995, 2001
- [44] engineering design for data networks” Global Telecommunications Conference, 1999. GLOBECOM '99
- [45] Scheduling” in Proc. of 10th IEEE Tyrrhenian International Workshop on Digital Communications: Multimedia Communications, Ischia, Italy, September, 1998
- [46] Mechanism in Multi-Radio Multi-Channel WMN”. The 10<sup>th</sup> International Symposium on Pervasive Systems, Algorithms and Networks, IEEE ISPAN'09, Page(s): 296 – 302. Dec 2009.
- [47] NS2.33, <http://www.isi.edu/nsnam/ns/>
- [48] Hyacinth NS2.1b9, [www.ecsl.cs.sunysb.edu/multichannel/](http://www.ecsl.cs.sunysb.edu/multichannel/)
- [49] g Protocol in NS2” Dept. of Information and Communications Engineering, University of Murcia, <http://masimum.dif.um.es/nsrt-howto/html/>, December, 2004.
- [50] Ramon A.C & Jesus P.C Interface Support in NS-2” University of Cantabria, Technical Reports, January, 2007. <http://personales.unican.es/aguero/files/ucMultiIfacesSupport.pdf>

- [51] NS2 Notebook: Multi-channel Multi-interface Simulation in NS2,  
<http://www.cse.msu.edu/~wangbo1/ns2/nshowto8.html>
- [52] NOAH & Hyacinth : Multi-Channel & Multi-Interface Extension in NS2  
<http://dobbylove.wordpress.com/2008/07/05/multi-channel-multi-interface-extension-in-ns2-and-noah-routing-hyacinth/>
- [53] “Tutoriel awk” par nyal. 2005, <http://nyal.developpez.com/tutoriel/gawk/gawk.pdf>
- [54] VERSION 0.1. 13 aout 2003,  
Université de Haute-Alsace. Laboratoire de Gestion des Risques et Environnement  
[www.gfcombustion.asso.fr/documents/gnuplot-intro.ps](http://www.gfcombustion.asso.fr/documents/gnuplot-intro.ps)
- [55] Differentiated Services Implementation”, Open IP , Nortel Networks, July, 2000.  
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.58.8500&rep=rep1&type=pdf>

# Les annexes

# A

---

## Annexe 1 : Le simulateur NS

---

### 1. Introduction

NS est un outil logiciel de simulation de réseaux informatiques. Il est principalement bâti avec les idées de la conception par objets, de réutilisation du code et de modularité. Il est devenu aujourd'hui un standard de référence en ce domaine. C'est un logiciel dans le domaine public disponible sur l'Internet. Son utilisation est gratuite. Le logiciel est exécutable tant sous Unix que sous Windows.

Cet annexe a pour but d'exposer les composantes de base de NS-2 en particulier ceux qui sont importants pour la mise en œuvre de notre protocoles

### 2. Les composants standard

#### Noeuds

Les noeuds sont fondamentaux dans une simulation. Ils effectuent le traitement et l'acheminement des paquets, et sont donc les entités les plus importantes parmi tous les composants réseau de ns-2. La Figure 5.1 montre la structure interne d'un tel noeud.

Un noeud est connecté à un canal sans fil. Il est construit à partir des parties suivantes:

- Un classificateur adresse : Il est utilisé pour la classification de paquets vers le port ou l'agent de routage.
- Un classificateur port, ainsi, passe les paquets à l'agent correspondant.
- Un agent de routage.
- Une couche de liaison responsable de la conversion d'adresse réseau en adresses matérielles à l'aide d'un module ARP.
- Une file d'attente d'interface utilisée pour le stockage et l'ordonnancement des paquets à l'égard de certains politiques bien définies.

- Une interface réseau qui envoie et reçoit les paquets sur le canal sans fil.
- Un modèle de propagation des ondes radio pour déterminer la puissance du signal de paquets reçus, et Par conséquent, si un paquet peut être reçu par une interface réseau ou non.
- un canal sans fil sur lequel les paquets sont distribués.

### **Paquets**

Les paquets sont l'unité d'échange entre les objets dans une simulation. Ils sont constitués de l'en-tête de paquets ce qui correspond à des protocoles différents qui peuvent être utilisés, et les données de paquet. Un nouveau protocole peut ajouter leur propre type d'en-tête à ceux disponibles.

### **Agents**

L'agents représentent des points de terminaison où les paquets sont produites ou consommées, et sont utilisés pour implémenter des protocoles à divers niveaux. Les agents de routage et les puits de trafic sont quelques exemples des agents qui sont fréquemment utilisées dans les simulations.

## **3. Interpréteur Tcl**

NS est un outil de simulation de réseaux de données. Il est bâti autour d'un langage de programmation appelé Tcl dont il est une extension. Du point de vue de l'utilisateur, la mise en œuvre de ce simulateur se fait via une étape de programmation qui décrit la topologie du réseau et le comportement de ses composants, puis vient l'étape de simulation proprement dite et enfin l'interprétation des résultats. Cette dernière étape peut être prise en charge par un outil annexe, appelé nam qui permet une visualisation et une analyse des éléments simulés. Dans cette section, nous allons montrer comment utiliser le simulateur avec les composants disponibles dans la distribution.

Une simulation est définie par un script OTcl. La réalisation d'une simulation implique la création et l'exécution d'un fichier avec une extension ". TCL". Un script Tcl :

- Définit une topologie de réseau (y compris les nœuds, les liens, et les algorithmes d'ordonnement et de routage d'un réseau).
- Définit un modèle de trafic (par exemple, le temps le de début et d'arrêt d'une session FTP).
- Collecte des statistiques et renvoie des résultats de la simulation. Les résultats sont habituellement écrits dans des fichiers de trace.

# B

---

## Annexe 2 : Support du multi interface dans NS2.33

---

### 1. Introduction

Pour rendre le simulateur NS capable de manipuler des nœuds avec plusieurs interfaces réseaux et permettre à un nœud de changer l'attribution du canal pour ses interfaces réseaux, nous devons changer les procédures de création et de configuration d'un nœud, les procédures de manipulation des canaux et des interfaces ainsi que les procédures de définition de protocole de routage.

Dans cette annexe, nous détaillons les changements nécessaires à effectuer sur ces procédures. Pour avoir le code complet des procédures modifiés, vous référer à l'adresse URL suivant : <http://telecom.inescporto.pt/~rcampos/ucMultiIfacesSupport.pdf>

### 2. Changement du code TCL

Les fichiers TCL assurent la creation et la configuration d'un noeud mobile dans NS sont: `tcl/lib/ns-lib.tcl` et `tcl/lib/ns-mobilenode.tcl`.

#### 2.1. Changement de `ns-lib.tcl`

Nous créons quatre nouvelles procédures. La première « `change_numifs` » est appelé avant la création du nœud et permet à l'utilisateur de spécifier le nombre d'interfaces réseaux d'un nœud donné. Cette procédure est appelé depuis le script de scénario et demande comme paramètre le nombre d'interface réseau.

La deuxième « add\_channel » permet l'ajout d'un canal à une interface réseau donnée. De même, cette procédure est appelé dans le script de scénario et nécessite comme paramètre l'index d'interface réseau et le canal.

La troisième « get\_numifs » n'est pas appelé dans le script de scénario, mais elle est sollicité par d'autre procédure pour savoir le nombre d'interface réseau que dispose un nœud.

La dernière procédure crée est « ifNum » dont on spécifie que le nœud qui doit être créer comporte plusieurs interfaces réseau. Cette procédure est utilisée comme argument dans la commande de configuration d'un nœud « node\_config ».

En plus de la création de ces procédures, deux autres procédures doivent être changés. La première est « node\_config » dans laquelle nous initialisons la variable **chan** comme une simple variable en cas d'une seule interface ou comme un tableau dans le cas de plusieurs interfaces. En plus nous ajoutons la variable « numifs » à la liste des argument de la commande.

La deuxième procédure qui doit être encore modifiée est « create\_wireless\_node ». Dans le cas où un nœud dispose de plusieurs interfaces, la procédure « add\_interface » qui est définit dans tcl/lib/ns\_mobilenode.tcl doit être appelée autant de fois que le nombre d'interfaces réseaux du nœud.

## 2.2. Changement de ns-mobilenode.tcl

Dans ce fichier, nous créons une seule procédure et nous modifions quelques procédures déjà existantes. La seule procédure à créer est « change\_interface » qui sert à changer l'attribution du canal pour une interface donnée. Cette procédure nécessite comme paramètre l'interface qu'on veut changer son canal et le nouveau canal, elle est appelé depuis le programme utilisateur ou le fichier script de scénario.

La première procédure à changer « add\_target », dans cette procédure nous interrogeons si le nœud dispose de plusieurs interfaces réseaux alors la commande « new\_if\_queue » doit être appelée autant de fois que le nombre d'interfaces réseaux du nœud.

La deuxième procédure modifiée est « add\_target\_ragent » qui est appelé dans la première. Quand l'agent est attaché au port RT\_PORT et il s'agit d'un agent de routage. Nous travaillons toujours avec la variable « numifs » dans ce cas pour lier l'agent de routage aux différentes couches liaisons correspondantes.

La dernière procédure qui doit être modifiée est « add\_inteface » pour la création de plusieurs tables ARP, une table pour chaque interface réseau en plus.

### 3. Changement du code C++

La plupart des changements ont été fait dans l'implémentation TCL, Toutefois, certaines modifications doivent être faites également dans les fichiers C++

#### 3.1. Changes on mobilenode.[cc,h]

Après la creation de la structure d'un noeud avec plusieurs interfaces en TCL, il est nécessaire de lui associer correctement le canal approprié. Le simulateur contrôle les nœuds qui sont connectés à un canal par le biais d'une liste qui est gérée par deux pointeurs (un pour les nœuds précédents et l'autre pour les noeuds suivants sur la liste). Ces pointeurs sont des variables simples à l'origine, mais dans le cas d'utilisation de plusieurs interfaces donc plusieurs canaux nous devons changer la déclaration de ces pointeurs comme des tables de pointeurs, une case pour chaque canal.

#### 3.2. Changes on channel.cc

Les deux tableaux mentionnés avant sont utilisés dans le fichier 'commun/channel.cc' pour gérer la liste de nœuds correspondant (jointure d'un nouveau nœud à un canal, la mise à jour). Afin de référer la liste appropriée, l'indice du canal correspondant doit utiliser dans la manipulation des deux tableaux.

#### 3.3. Changes on mac-802\_11.cc

Avec l'utilisation de plusieurs interfaces dans chaque nœud, le protocole de la couche mac, dans notre cas 802.11 doit être au courant du ce nombre. Les modifications nécessaires pour cela sont trouvés dans le lien suivant : <http://www.cse.msu.edu/~wangbo1/ns2/nshowto8-6.html>

#### 3.4. Changes on arp.cc

De même pour le module ARP de chaque nœud. Nous modifions "arp.cc" pour éviter toute erreur de mise à jour de la table ARP des nœuds voisins. Vous pouvez trouver les principaux changements dans ses deux liens : <http://www.cse.msu.edu/~wangbo1/ns2/nshowto8-3.html> et <http://dobbylove.wordpress.com/2008/07/05/multi-channel-multi-interface-extension-in-ns2-and-noah-routing-hyacinth/>

#### 4. Changement dans le code du protocole de routage

Le but final de l'implémentation du modèle multi-interfaces est sans doute son utilisation. Par conséquent, l'architecture des agents externe du simulateur doit être changée afin d'utiliser cette nouvelle fonctionnalité. Dans cette section, nous montrons comment un agent de routage doit être adapté pour utiliser des nœuds avec plusieurs interfaces, de sorte qu'il est possible de bénéficier du modèle modifié.

##### 4.1. Changement dans l'implémentation de l'agent de routage

Puisque nous voulons que le nombre d'interfaces par nœud soit flexible et, en outre, nous voulons maintenir le comportement d'héritage du simulateur, il est nécessaire que l'agent de routage permette de suivre le nombre d'interfaces qu'il gère. Un nouveau membre de la classe de routage agent, **nifaces**, est déclarée de manière à garder cette information. Le nombre d'interface sera défini à partir du script scénario, c'est pourquoi au début, sa valeur est fixée à 0 (ie dans le constructeur de l'agent)

Comme on le voit sur la figure 5.2, c'est l'agent de routage qui doit décider de l'interface de sortie dont il a besoin pour passer le paquet. Au lieu d'utiliser l'unique 'ifqueue' et 'target' traditionnelle que tout agent de routage peut l'utiliser, nous déclarons deux tableaux, '**targetlist**' et '**ifqueuelist**'. Le premier stocke les modules LL pour toutes les interfaces d'un nœud particulier, tandis que le deuxième permet de conserver leurs files d'attente correspondantes.

L'étape suivante serait la modification de la méthode « command » de la classe agent de routage de manière à initialiser la valeur de la variables précitées à partir du script Tcl et de pouvoir les utilisés.

Avec tous ces changements, il est effectivement possible de créer des nœuds avec plusieurs d'interface. L'étape suivante consiste à ajouter l'intelligence requise au sein de l'implémentation de l'agent de routage, de sorte qu'il peut décider de l'interface qui doit transmettre chaque paquet. D'autre part, il est également bien connu que l'utilisation des transmissions de diffusion est tout à fait pertinente dans les protocoles de routage de réseau ad hoc (par exemple pendant le processus de découverte de route); lorsqu'il y a plus d'une interface disponible, un paquet de diffusion doit être transmis à travers toutes les interfaces d'un nœud.

D'autre part, pour les transmissions, un indice **Iface** est utilisé de manière à sélectionner la cible appropriée (l'entité LL de l'interface que le paquet doit être envoyé). Cette valeur doit

être soigneusement sélectionnée et doit être conservée dans la table de routage avec le reste des informations nécessaires. En ce sens, la méthode utilisée pour créer une nouvelle entrée dans la table de routage doit être actualisée de manière à indiquer l'interface de sortie qui doit être utilisé pour atteindre la destination.

#### **4.2. Changement dans la table de routage**

L'information de l'indice d'interface doit être stockée dans l'entrée de la table de routage pour la destination correspondante afin qu'elle puisse être utilisée par les futures transmissions, c'est-à-dire, pour transmettre un paquet à une destination, il ne suffit pas de connaître le prochain saut seulement, mais l'agent de routage doit être conscient de l'interface de sortie, qu'il a besoin d'utiliser pour atteindre la destination. Par conséquent, une nouvelle variable '**interface**' doit être ajoutée à la définition d'entrée de la table de routage.

---

## Annexe 3 : Description des messages échangés

---

### 1. Introduction

Nous donnerons, dans cette annexe, une description détaillée des structures de données utilisés pour présenter les ressources réseaux et l'agent de routage, ainsi que le détail des paquets de contrôle échangés pour la construction des arbres de routage et le maintient de l'équilibrage de charge durant la simulation

### 2. Structure de données

#### 2.1. Classe Binding

La classe 'Binding' résume l'état des interfaces de chaque noeud, donc pour manipuler les ressources ou les interfaces, on utilise cette classe qui a les champs suivants

Champ	Description
Chnl	Représente l'ID du canal assigné à cette interface
initialized	Boulean (0,1)
type	Type de cette interface (parent, enfant)
traffic_classe	classe de trafic véhiculé par cette interface.
chnl_avg_usage	L'utilisation du canal par cette interface.
nic_avg_send_usage	Ces champs sont utilisés pour le calcul de la charge des interfaces.
nic_avg_rcv_usage	
bytes_just_sent	
bytes_just_rcv	
last_pointer	
bytes_just_sent[Measurement_period]	

bytes_just_recv[Measurement_period]	
num_children	Nombre d'enfants affectés à cette interface.
children[max_child_per_nic]	Table d'ID des noeuds enfants affectés à cette interface
child_to_move	Ces champs sont utilisés dans le cas de réattribution de canal.
nic_to_move_to	
chnl_to_move_from	
ack_table[NUM_NODES]	
prev_chnl	
recently_adjusted	

## 2.2. Classe SpanningTree

A chaque noeud dans le réseau est affecté un agent de routage qui a la structure suivant:

Champ	Description
ra_addr_	Adresse du noeud
gwnode	Boulean, indique si ce noeud est une passerelle.
num_backups_be	Nombre de chemin de secoure pour l'arbre best effort
backup_be[NUM_NEIGHBOR]	Table des chemins de secoure pour l'arbre best effort
num_backups_qos	Nombre de chemin de secoure pour l'arbre QoS
backup_qos[NUM_NEIGHBOR]	Table des chemins de secoure pour l'arbre QoS
rtable_be	Table de routage de l'arbre best effort
rtable_qos	Table de routage de l'arbre QoS
ifaces	Nombre d'interfaces de ce noeud
ration_qos_be	% ressource dédiée pour chaque classe, se met a jour périodiquement
targetList[max_iface]	Utile pour le support de multi interface.
ifqueuelist[max_iface]	Utile pour le support de multi interface.
nic[NUM_NIC]	Table des interfaces de type classe Binding
control	L'interface de contrôle
parents_be[NUM_NIC]	Les interfaces parents dans l'arbre de trafic best effort

parents_qos[NUM_NIC]	Les interfaces parents dans l'arbre de trafic QoS
child_be[NUM_NIC]	Les interfaces enfants dans l'arbre de trafic best effort
child_qos[NUM_NIC]	Les interfaces enfants dans l'arbre de trafic QoS
newparent	Adresse de nouveau parent
hop_map[NUM_NODES]	Table de structure Next_Hop (détailler par la suite)
usage_map[NUM_NODES][NUM_CHNL]	Table de structure Usage (détailler par la suite)
chnl_map_be[NUM_CHNL]	Charge de trafic best effort dans chaque canal
chnl_map_qos[NUM_CHNL]	Charge de trafic QoS dans chaque canal
chnl_hop[NUM_CHNL]	Priorité de chaque canal

### Structure Next\_Hop

Cette structure représente les interfaces utilisés entre le noeud en question et ses noeuds voisins, elle est décrit par le tableau suivant

Champ	Description
num_nics	Nombre d'interface dedie à ce saut
nic[NUM_NIC]	Table des interfaces qui relie ce noeud avec son voisin
nic_ctr	L'interface utilisé en dernier
status	L'état de ce saut (en changement, stable)

### Structure Usage

Cette structure permis au noeud de savoir l'état d'utilisation (la charge) des autres noeuds aux autres canaux, elle contient les champs suivants:

Champ	Description
usage_be	La charge de trafic best effort
usage_qos	La charge de trafic avec QoS
hop_count	Distance en nombre de saut par rapport la passerelle.
seq_no	Numéro séquentiel

### Structure de l'entrée de la table de routage

Les éléments de la table de routage sont représentés par la structure de l'entrée de la table de routage, qui a les champs suivants:

Champ	Description
dest	destination
next_hop	Saut suivant vers le noeud filaire
nic_id	L'ID d'interface réseau assigné au lien (noeud, dest)
next_to_gw	Boulean indique si ce noeud est voisin direct de la passerelle.
metric hop_count	Distance vers la passerelle.
metric avail_bw	La bande passante disponible
path	Le chemin vers la passerelle.
seq_no	Numéro séquentiel
status	L'état de cette entrée (stable, temporaire ou en changement)

Chaque noeud dans le réseau a deux tables de routage, qui décrit ses sous arbres de routage pour chaque classe de trafic.

### 3. Étape initialisation

Au démarrage, chaque noeud,

- Ajoute une entrée pour le noeud filaire dans ses tables de routage avec une métrique indéfini. Seul le noeud passerelle qui peut défini cette métrique.
- Assigne des canaux différents pour toutes ses interfaces de façon aléatoire (car initialement, tous les canaux et tous les interfaces ont la même charge).
- Différencie entre les interfaces qui vont être utilisé par l'arbre de routage QoS de ceux qui vont être utilisé par l'arbre de routage BE, en se basant sur un facteur initial k (on propose au départ, qu'il aura plus de trafic QoS que de trafic BE).

## 4. Structure des paquets de contrôle

### 4.1. Paquet de construction de l'arbre

Périodiquement, chaque noeud publie à l'attention de tous ses voisins a un saut

Champ	Description
type_message	Type de paquet de contrôle (ST_CONST)
type_traffic	Classe de trafic (best effort ou QoS)
hop_count	Distance par rapport la passerelle

Dst	dst de WIRED-HOST dans la table de routage
Hop	hop de WIRED-HOST dans la table de routage
avail_bw	bande passante disponible
next_to_gw	Ce noeud est il voisin direct de la passerelle (1=oui, 0=non)
seqnum	Numéro séquentiel
path	Chemin vers la passerelle.

#### 4.2. Paquet de demande de jointure/quitte de l'arbre

Ce message est utilisé dans quatre cas, pour construire le message

1. ST\_JOINT: Quand un nœud reçoit le paquet ST\_CONST, et si la métrique de routage de ce paquet est meilleure, il répond par un paquet ST\_JOINT pour demander la jointure a l'arbre de routage enraciné par l'expéditeur. Pour éviter les conflits qui peuvent surgir dans l'attribution de canaux entre les interfaces parents, on distingue deux cas:

1. Si ce noeud va envoyer un message ST\_JOINT pour l'arbre qos, il lui ajoute des informations sur les canaux déjà attribués a ses parents be.
2. Dans l'autre cas, ou le message ST\_JOINT est envoyé pour l'arbre BE, alors il ajoute des informations sur les canaux déjà attribués aux interfaces de l'arbre qos (parents et enfants) pour permettre a l'arbre QoS d'être plus stable.

Donc, le nœud parent lorsque il reçoit ce message, il doit éviter ces canaux quand il cherche l'interface la moins chargée.

2. ST\_LEAVE: Quand un nœud reçoit le paquet ST\_CONST, et trouve que la métrique de routage de ce paquet est meilleur, il envoi le paquet ST\_LEAVE a son actuel parent pour qui se détache de son arbre.

3. ST\_ADDCH:

4. ST\_DELCH:

Champ	Description
type_message	Type de paquet de contrôle (ST_JOINT, ST_LEAVE, ST_ADDCH, ST_DELCH).
type_taffic	Classe de trafic (best effort ou QoS)
Seqnum	Numéro séquentiel
Added_cost	Le cout ajouter après la jointure de ce noeud

nb_children	Nombre d'enfants pour le sous arbre enraciner par ce noeud
children[MAX_CHILD]	La liste des enfants
counter[MAX_CHILD]	Liste de counter correspondant a chaque enfants
chnl_parents[NUM_PARENT_NIC]	Liste des canaux déjà assignés aux autres parents
chnl_child[NUM_NIC]	Liste des canaux déjà assignés aux enfants de l'arbre QoS
nb_nic_child	Nombre d'interface enfants QoS

#### 4.3. Paquet de réponse à la demande de jointure

Après recevoir le paquet ST\_JOINT, un nœud répond par ce paquet

Champ	Description
type_message	Type de paquet de contrôle (ST_ACCEPT)
type_taffic	Classe de trafic (best effort ou QoS)
hop_count	Distance par rapport la passerelle
avail_bw	bande passante disponible
next_to_gw	Ce noeud est il voisin direct de la passerelle (1=oui, 0=non) <b>a vérifier</b>
seqnum	Numéro séquentiel
Chnl	L'id de canal qui sera attribué pour lien entre le noeud et son parent
path	Chemin vers le noeud filaire.

#### 4.4. Paquet d'état de la bande passante disponible

Ce paquet est envoyé récursivement, de père en père, après la jointure ou la départ d'un enfant. Il sert a mettre a jour la bande passante disponible chez la passerelle.

Champ	Description
type_message	Type de paquet de contrôle (UPDATE_STATS)
type_taffic	Classe de trafic (best effort ou QoS)
parent_nic_avg_send_usage	Charge de trafic transmit par cette interface.
parent_nic_avg_rcv_usage	Charge de trafic reçu par cette interface.
source	L'ID du noeud source de ce paquet
Action	L'action qui sera déroulé (ajout ou suppression)

#### 4.5. Paquet de mise à jour de l'utilisation des canaux

Ce paquet est envoyé, périodiquement, dans chaque évènement de mesure ou d'équilibrage de charge. Un noeud transmis par ce paquet les canaux qu'il utilise dans les deux arbres avec l'état d'utilisation.

Champ	Description
type_message	Type de paquet de contrôle (CHNL_UPDATE)
From	L'id du noeud source
Relay	Nombre de saut dans la zone d'interférence
chnl_seq_no	Numéro séquentiel
myhopcount	Distance par rapport la passerelle
chnl_be[NUM_NIC]	Les ids des canaux pour les différentes interfaces dans l'arbre BE
chnl_usage_be[NUM_NIC]	Les charges des canaux pour les interfaces dans l'arbre BE
chnl_qos[NUM_NIC]	Les ids des canaux pour les différentes interfaces dans l'arbre QoS
chnl_usage_qos[NUM_NIC]	Les charges des canaux pour les interfaces dans l'arbre QoS

#### 4.6. Paquet de changement de réattribution de canal

Ce paquet permet de changer l'attribution de canal pour une interface donnée. Il est envoyé par un parent a son enfants pour équilibrer la charge des canaux

Champ	Description
type_message	Type de paquet de contrôle (CHNL_CHANGE)
From	L'id du noeud source
prev_chnl	L'id du canal précédant
new_chnl	L'id du nouveau canal
nic_id	L'id de l'interface en question

### 5. Les messages de contrôles et les traitements associés

La liste des messages de contrôle est la suivante: ST\_SOLICITE, ST\_CONST, ST\_JOINT, ST\_LEAVE, ST\_ACCEPT, CHNL\_UPDATE, UPDATE\_STATS, CHNL\_CHANGE, ST\_ADDCH, ST\_DELCH. ST\_JOINT, ST\_LEAVE

**ST\_CONST**

Quand un noeud reçoit ce message, il

1. Ajoute l'expéditeur dans la liste des parents de secours
2. S'il le reçoit pour la première fois, il l'accepte comme parent et lui répond par un message ST\_JOINT
3. Sinon, il le compare avec le parent courant et choisit le meilleur

**ST\_JOINT**

A la réception de ce message, un noeud réagit de la manière suivante

1. Afin d'équilibrer la charge entre les interfaces, en cherche à choisir l'interface la moins chargée en terme de charge de trafic sur tous les enfants affectés à cette interface pour l'assigner à l'enfant expéditeur, tout en évitant celle assignée avec les canaux envoyés en paramètres.
2. Ajouter les descendants de cet expéditeur à la table de routage
3. Répondre par un message ST\_ACCEPT
4. Relancer l'évènement de l'équilibrage de charge
5. Envoyer le message ST\_ADDCH au parent pour ajouter cet enfant dans leur table de routage.

**ST\_ACCEPT**

Quand un noeud reçoit un tel message, il fait le traitement suivant

1. Si l'expéditeur du message est différent du parent actuel alors il envoie un message ST\_LEAVE au parent actuel
2. Assigne la première interface au lien qui relie ce noeud avec son parent.
3. Attribue le canal envoyé en paramètre dans le message ST\_ACCEPT au lien du parent.
4. Demande au nouveau parent de mettre à jour l'état de la bande passante disponible après la jointure de ce noeud, par l'envoi du message UPDATE\_STATS
5. Voir s'il y a un conflit avec les canaux assignés aux noeuds enfants et relancer l'évènement de l'équilibrage de charge en signalant le conflit trouvé.

**ST\_ADDCH**

Après recevoir le message ST\_ADDCH, un noeud

1. Ajoute les noeuds de la liste des enfants envoyés en paramètre dans sa table de routage,
2. Renvoie le message ST\_ADDCH à son parent, jusqu'à l'arrivée au noeud passerelle.

**ST\_LEAVE**

A la réception de ce message, un noeud fait le traitement suivant

1. Mètre à jour la liste des interfaces enfants, en enlevant le noeud expéditeur de cette liste,
2. Enlever le noeud expéditeur avec ses descendants de la table de routage,
3. Envoi le message ST\_DELCH à son parent.
4. Relancer l'évènement de l'équilibrage de charge après le départ de cet enfant.

**ST\_DELCH**

Quand un noeud reçoit ce message, il

1. Enlève les noeuds de la liste des enfants envoyés en paramètre de sa table de routage,
2. Renvoi le message ST\_DELCH a son parent, jusqu'au arrivé au noeud passerelle.

**UPDATE\_STATS**

1. Si le noeud récepteur n'est pas une passerelle, il renvoi ce message a son parent de façon récursive, jusqu'à arrivé au noeud passerelle, et met à jour la charge de trafic de ses interfaces parents par l'ajout ou la soustraction de la charge transmise dans le message suivant l'action voulu (addition ou suppression).
2. Sinon, le noeud passerelle met à jour la charge de trafic de son interface enfant qui référence l'enfant expéditeur par l'ajout ou la soustraction de la charge transmise dans le message suivant l'action voulu.
3. le noeud passerelle, relance l'évènement de reconstruction de l'arbre de routage à cause de changement de la charge de trafic.

**CHNL\_UPDATE**

A la réception de ce message, un noeud

- Met à jour l'entrée de l'expéditeur dans la table usage\_map[nodes][chnls] par la valeur 'usage' transmis en paramètre.
- S'il est dans la zone d'interférence du noeud expéditeur, il renvoi ce message à tous ses voisins directs.

**CHNL\_CHANGE**

- Change l'attribution de canal suivant les paramètres envoyés dans le paquet,
- Diffuse le paquet CHNL\_UPDATE dans la zone d'interférence,
- Voir s'il y a un conflit avec les canaux déjà assignés aux interfaces enfants et relancer l'évènement d'équilibrage de charge pour le régler.

# D

---

## Annexe 4 : Fichier de trace et le langage AWK

---

### 1. Fichier de trace

Le fichier ascii contenant les traces d'une simulation NS est organisé en ligne de 12 champs comme le montre la Figure D.1.

Evénement	Temps	Du noeud	Vers le noeud	Type paquet	Taille paquet	Flags	FID	Adresse SRC	Adresse DST	Numéro séquence	ID paquet
-----------	-------	----------	---------------	-------------	---------------	-------	-----	-------------	-------------	-----------------	-----------

Figure D.1 : Format d'une trace NS

La signification de chacun de ces champs est la suivante :

1. Le premier champ indique le type de l'évènement. On peut y trouver 6 symboles r, +, -, d, s et f qui signifient respectivement que le paquet a été reçu (la sortie du lien), a été mis en attente dans une file, a quitté la file d'attente, a été supprimé de la file (et donc perdu) et subit une erreur (une erreur binaire), a été transmis et a été forwerdé. Ainsi, le temps d'attente se mesure entre un + et -, le temps de transmission combiné au temps la propagation entre - et r et enfin le délai de traitement s'il existe entre r et +.
2. Le deuxième champ indique l'instant auquel l'évènement s'est produite.
3. Ce champ indique le noeud source sur le lien
4. Ce champ indique le noeud destination sur le lien
5. Ce champ indique le type du paquet (par exemple CBR ou TCP)
6. Ce champ indique la taille du paquet
7. Ce champ indique quelques flags
8. Ce champ indique l'identifiant du flot. Cette identifiant peut être fixé dans le script TCP. Utile pour l'analyse et pour colorer les flots avec NAM.

9. Ce champ indique l'adresse source désignée dans le format "noeud.port".
10. Ce champ indique l'adresse destination (même format).
11. Ce champ indique le numéro de séquence du paquet affecté par la couche réseau.
12. Ce dernier champ indique un identifiant unique du paquet.

## 2. Généralités sur Awk

Awk est un programme qui permet de chercher dans des fichiers des lignes ou portion de texte contenant un ou des patrons et d'y effectuer des opérations. Il y a au moins deux façons de lancer awk.

- Directement le lancer depuis la ligne de commande : `awk <script awk> tr.out` ou
- En mémorisant les commandes awk dans un script : `cat tr.out | awk -f <script awk>`. Note, Pour passer un argument en ligne de commande à awk, on utilise l'option `-v <nom variable> <valeur variable>`.

Quand awk lit une ligne d'un fichier, il divise la ligne en champs basés sur le séparateur de champs en entrée, FS, qui est une variable awk. Cette variable est prédéfinie avec un ou plusieurs espaces et tabulations. Les variables \$1, \$2, \$3, ..., \$N stockent les valeurs du premier, second, troisième jusqu'au dernier champs de la ligne traitée. La variable \$0 stocke la valeur de la ligne entière.

Un programme awk se compose comme une suite de blocs de code compris entre { } qui sont appliqués aux lignes de données du fichier si un patron est vérifié. Voici quelques types de patron que l'on peut avoir :

**Tableau D.1 : Principaux blocs d'un programme AWK**

BEGIN	Le bloc qui suit est exécuté au début du programme, une seule fois.
END	Le bloc qui suit est exécuté à la fin du programme, une seule fois.
/expression régulière/	Le bloc qui suit est exécuté si les données en cours de traitement correspondent à l'expression régulière. Par exemple, /#/ Permet de sélectionner les lignes commençant par un #.
expression relationnelle	Le bloc qui suit est exécuté si l'expression est vraie. Par exemple <code>mvar=5</code>
modèle && modèle	Le bloc qui suit est exécuté si les deux modèles sont vérifiés.
modèle    modèle	Le bloc qui suit est exécuté si au moins un des modèles est vérifié.
modèle0 ? modèle1 : modèle2	Le bloc qui suit est exécuté si les modèles 0 et 1 sont vérifiés ou si le modèle2 uniquement est vérifié
! modèle	Le bloc qui suit est exécuté si le modèle n'est pas vérifié.
modèle1, modèle2	Le bloc qui suit est exécuté pour la partie des données en cours commençant par modèle1 et finissant par modèle2.