Conception et évaluation d'un modèle adaptatif pour la qualité de service dans les réseaux MPLS

Khodor Abboud

N° d'ordre : | 1 | 4 | 8 |

ÉCOLE CENTRALE DE LILLE

THÈSE

présentée en vue d'obtenir le grade de

DOCTEUR

en

Spécialité : Automatique et Informatique Industrielle

par

Khodor ABBOUD

Doctorat délivré par l'École Centrale de Lille

Conception et évaluation d'un modèle adaptatif pour la qualité de service dans les réseaux MPLS

Soutenue le 20/12/2010 devant le jury d'examen :

Président	Bernard RIERA	Professeur à la faculté des sciences de REIMS
Rapporteur	Bernard RIERA	Professeur à la faculté des sciences de REIMS
Rapporteur	Ouajdi KORBAA	Professeur à l'Université de Sousse (Tunisie)
Membre	Aziz NAKRACHI	Maître de Conférences à l'Ecole Polytechnique de Lille
Membre	Pierre-Alain YVARS	Maître de conférences-HDR, LISMMA, SupMéca
Directeur de thèse	Armand TOGUYENI	Professeur des universités, LAGIS, Ecole Centrale de Lille
Directeur de thèse	Ahmed RAHMANI	Maître de Conférences-HDR, LAGIS, Ecole Centrale de Lille

Thèse préparée au sein du Laboratoire LAGIS

Équipes MOCIS/STF

École Doctorale SPI 072

A mes parents,

A mes soeurs et mes frères,

pour leur amour et leur soutien,

Remerciements

En premier lieu, je tiens à exprimer ma profonde gratitude aux directeurs de cette thèse le Professeur Armand TOGUYENI et à Ahmed RAHMANI, Maître de Conférences - HDR. Je veux les remercier pour leur encadrement plein d'enthousiasme et de rigueur et pour la confiance dont ils ont fait preuve à mon égard.

Ensuite, je souhaite témoigner de mes remerciements chaleureux au Professeur Bernard RIERA de m'avoir accordé l'honneur de présider le jury d'examen.

Puis, je tiens à exprimer ma vive gratitude au Professeur Bernard RIERA et au Professeur Ouajdi KORBAA pour l'intérêt dont ils ont fait preuve à l'égard de mon travail en acceptant d'être les rapporteurs de cette thèse.

Je désire également remercier Aziz NAKRACHI, Maître de Conférences à l'université de Lille1, et Pierre-Alain YVARS, Maître de Conférences - HDR au SUPMECA, d'avoir accepté d'examiner ce travail de recherche et de faire partie du jury.

Finalement, ces remerciements ne seraient pas complets si je n'y associais pas toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce travail, en particulier, mes collègues, tout le personnel du LAGIS, et de l'Ecole Centrale de Lille pour leur bonne humeur et leur disponibilité.

Table des matières

Table des figures

Introduction générale

Contexte générale

Depuis quelques années, nous assistons à un développement rapide des applications Communicantes. En effet, l'intérêt grandissant que suscitent les réseaux, notamment le réseau Internet, l'avènement des nouvelles technologies de l'information et les nombreuses applications (les services de téléphonie, la vidéoconférence, le peer-to-peer, la télé robotique, etc) font que le nombre d'utilisateurs (particuliers, entreprises, laboratoires, etc) est sans cesse croissant. Les nouveaux services offerts utilisent des techniques et des infrastructures existantes, qui ne garantissent pas souvent à ces utilisateurs un fonctionnement correct ni une qualité de prestation acceptable. En d'autres mots, ces infrastructures ne répondent plus aux exigences de certaines applications en terme de qualité de service. Ainsi, l'utilisation des nouvelles générations de réseau dans le cadre d'applications multimédia ou de services à qualité garantie, impose que l'acheminement soit assuré avec une Qualité de Service (QdS) maîtrisée. La qualité de service (QdS) comporte une série de paramètres qui permettent de caractériser les garanties qui peuvent être fournies à chaque flux (ou connexion) transitant par le réseau. Ces paramètres sont généralement la bande passante, le délai de bout en bout, la gigue ou le taux de perte des données. L'Internet, initialement conçu pour échanger

de simples données (mail, FTP, etc), est devenu un réseau universel qui sert de support à de plus en plus d'applications distribuées plus exigeantes par rapport à des contraintes de temps et sont de plus en plus gourmandes en ressources réseau (particulièrement en bande passante) comme par exemple les applications multimédias (audio et vidéo streaming, etc) et les applications de contrôle/commande (la télé opération, etc).

Cependant, le problème de l'intégration de la qualité de service dans les réseaux constitue un vaste sujet de recherche et a fait l'objet de nombreuses techniques proposées dans la littérature. Celles-ci interviennent à différents niveaux. Le premier niveau concerne les noeuds extrêmes d'un lien de communication (noeuds d'entrée, noeuds de sortie) avec comme exemples le démarrage lent et le contrôle de débit dans le protocole TCP, tandis que le deuxième niveau concerne quant à lui les noeuds intermédiaires du réseau. Ainsi, des techniques de prévention de congestion (RED, WRED) et des solutions de gestion explicite de la Qualité de service réseau (IntServ/RSVP, DiffServ) ont été proposées. Même si ces techniques apportent des éléments de solution, elles ne garantissent pas à elles seules la QdS de bout en bout et doivent être complétées par des techniques de routage adaptatif.

Tout a commencé au début des années 90 au sein de l'IETF, où des groupes de recherche travaillant sur la qualité de service IP ont vu le jour. Ces groupes ont tenté de développer des mécanismes de plus en plus sophistiqués, tout d'abord pour l'optimisation du partage de ressources et le contrôle du trafic et par la suite, pour la gestion des files dans les routeurs IP. Le premier réseau paquet large bande à offrir une QdS à ses utilisateurs et à proposer plusieurs classes de services, afin de traiter différemment les flux, est l'ATM (Asynchronous Transfer Mode). Ce réseau à cellules commutés permet de créer des chemins, de bout en bout et ainsi servir les paquets de chaque flux en

fonction de sa classe de service d'appartenance. Toutefois ce réseau présente quelques inconvénients dont les plus fréquemment cités sont l'introduction d'un surplus d'entête de 5 octets toutes les 48 octets utiles formant la cellule ATM, ou l'ajout d'un nouveau plan de contrôle et de gestion dans le réseau. Une autre technique de gestion de la QdS qui sera introduite plus tard par l'IETF est IntServ (Integrated Services). Cette technique permet de traiter les flux de paquets en fonction de la demande de la source juste avant de démarrer l'envoi des paquets utiles. IntServ ne nécessite pas de nouveau plan de contrôle mais uniquement le développement d'une signalisation adéquate RSVP (Resource ReserVation Protocol). Cependant, ce schéma se heurte à un autre problème, celui du facteur d'échelle. Avec IntServ, chaque routeur dans le réseau doit garder l'état de chaque flux qui y transite jusqu'au moment où la liaison s'achève. En remarquant que plusieurs milliers de flux peuvent passer sur un routeur d'un coeur de réseau, nous réalisons immédiatement pourquoi une telle technique ne peut être utilisée dans le coeur des réseaux haut débit. Par la suite, de nouvelles solutions intuitives ont été proposées et adoptées dans l'Internet, dont la plus importante est le principe de DiffServ. DiffServ (Differentied Service) est un mécanisme de différentiation de classes au niveau de chaque routeur. Il résout le problème du facteur d'échelle de IntServ en définissant un nombre limité de comportement PHB (Per Hop Behavior) au niveau de chaque noeud. Dans la même optique, l'Ingénierie de Trafic IT (Traffic Engineering TE) est apparue comme un autre moyen d'augmenter la qualité du service dans un réseau, en dimensionnant plus finement les ressources disponibles. Ce concept peut être lié au routage avec qualité de service ou à l'utilisation de la différenciation de services. Il s'agit d'adapter le comportement du réseau à chaque classe de service en modifiant le routage et/ou le traitement dans les équipements intermédiaires. A partir

de ces différentes constatations, MPLS (Multi Protocol Label Switching) a été conçu pour créer des chemins commutés dans le réseau IP d'une part, et pour pouvoir implémenter les avantages de ATM et ceux liés à l'ingénierie de trafic et à la qualité de service QdS d'autre part.

Ce type de réseau est caractérisé par l'hétérogénéité de ses liens, ainsi que par des conditions de trafic dynamiques, ce qui nécessite la prise en compte de la QdS au niveau du routage. Par conséquent, toute approche de routage adaptatif doit être suffisamment réactive et robuste pour prendre en compte toute modification des conditions de trafic tout en minimisant le temps d'acheminement de bout en bout. Ainsi, des approches de routage basées sur l'ingénierie de trafic avec des contraintes de qualité de service ont été proposées.

Par ailleurs, les performances de telles approches sont dépendantes de la qualité de la coopération entre ses différents éléments. Au fur et à mesure de l'évolution de l'Internet et de ses applications, les utilisateurs sont devenus de plus en plus exigeants et ont réclamé à leurs fournisseurs l'accès à des garanties de services (Service Level Agreement). Les utilisateurs et les opérateurs veulent donc connaître l'état actuel des performances du réseau et de la qualité de service fournie, ceci afin d'assurer les services garantis. Ces nouveaux besoins ont considérablement contribué à l'émergence d'un nouveau domaine de recherche qui est l'évaluation des performances. L'évaluation de performances permet de prévoir le comportement du réseau dans certaines situations, et en particulier de prévoir les éventuels problèmes et de trouver les solutions. L'évaluation de performances d'un réseau peut être étudiée en utilisant la simulation. En effet, la simulation possède l'avantage de pouvoir étudier des réseaux de tailles et de complexités différentes.

On peut également évaluer les performances d'un réseau grâce à des mé-

thodes analytiques. Le réseau et son comportement peuvent être étudiés en utilisant une modélisation mathématique et une résolution analytique. Cette approche permet de connaître la dynamique exacte du système lorsque des solutions existent. Le modèle mathématique peut aussi servir pour la modélisation de l'état du réseau à des finalités de contrôle ou de commande.

Objectifs de la thèse

Dans ce travail de thèse nous abordons dans un premier temps l'évaluation de performances des modèles de routage multi-chemins pour l'ingénierie de trafic et l'équilibrage de charge sur un réseau de type IP/MPLS (MPLS-TE). Nous comparons notamment la capacité de ces modèles à équilibrer la charge tout en faisant de la différentiation de trafic, en les appliquant sur de grandes topologies générées par le générateur automatique des topologies BRITE. Ces topologies s'approchent en forme et en complexité du réseau réel. Ceci afin de mesurer l'impact de leur complexité respective et donc la capacité à les déployer sur des réseaux de grande taille (scalabilité). Dans un second temps, nous proposons un concept de modélisation d'un réseau de commutations de paquets établi sur la base de la théorie différentielle de trafic. Ce modèle sert à estimer l'état de charge du réseau et permet de proposer ainsi des approches de contrôle de congestion et de commande sur l'entrée du réseau améliorant le plan de routage adaptatif et l'équilibrage de charge dans le réseau IP/MPLS.

Contributions de la thèse

Nous avons ciblé nos travaux sur les réseaux de type IP/MPLS incluant des mécanismes de l'ingénierie de trafic et de différenciation de service Diff-Serv. Les contributions de cette thèse concernent principalement trois parties:

La première concerne l'étude des différentes techniques et outils de mesure pour l'évaluation de performances des modèles de routage multi-chemins dans les réseaux IP/MPLS. Nous avons adopté la méthode d'évaluation par simulations, en utilisant le simulateur réseau NS-2. Nous avons proposé un modèle de simulations basé sur la topologie, le trafic et l'algorithme à simuler. Nous avons effectué une étude comparative des générateurs de topologies existants et retenu le générateur de topologies automatique BRITE. Afin d'adapter le format du fichier générée par BRITE en un format utilisé par les simulations, nous avons développé un script appelée "brite2ns". Ceci a permis d'analyser la scalabilité des modèles en augmentant la taille et la complexité du réseau, ainsi que l'effet du changement de topologie sur les performances des modèles. Ensuite, nous avons proposé un script contenant les différents paramètres et types de trafics envoyés dans le réseau. Ce script a permis de générer un trafic différencié selon plusieurs classes EF, AF et BE, ainsi que de réaliser plusieurs scénarios de charge du réseau entre faible, moyenne et charge élevée. Enfin, nous avons réalisé le script contenant le modèle à simuler. Il est basé sur les deux autres scripts de topologie et de trafic, et implémente les techniques de l'ingénierie de trafic et de routage multi-chemins dans le réseau. L'analyse des résultats de simulation a servi à l'évaluation des performances des deux modèles. L'objectif de cette première partie est de valider le passage à l'échelle des modèles concernées et en dépasser les problèmes techniques résultants.

La deuxième contribution consiste en la proposition d'un modèle adaptatif

basé sur la technique de routage explicite dans les réseaux de type IP/MPLS, pour l'estimation de l'état de charge du réseau et mesurer la bande passante disponible sur ses chemins de bout en bout. Le modèle proposé est établi sur la base de la théorie différentielle du trafic. Il repose sur les approches à compartiments et graphique de représentation des réseaux. Après un état de l'art sur les différents modèles existants, nous optons pour un modèle fluide basé sur le principe d'accumulation de flux dans la file d'attente d'un routeur. L'idée est d'établir un système d'équations différentielles basé sur des approximations dynamiques qui dépendent de l'état interne de chaque routeur et des interactions entre les routeurs transférant le trafic. Ce système exploite dynamiquement les informations et les mesures obtenues lors du transfert du trafic dans le réseau, afin de quantifier les paramètres caractérisant un goulot d'étranglement dans les chemins du réseau. Pour mettre en oeuvre ce système, deux étapes sont nécessaires: le calcul du flux de transfert dans le réseau, et la représentation des interactions entre ses différents composants. Nous avons commencé par développer la Méthode d'Assemblage Comportementale (MAC) basée sur l'approche à compartiments. Cette méthode a été utilisée pour calculer le flux de transfert à la sortie de chaque routeur en fonction du plan de routage explicite et de l'état interne de la file d'attente correspondante. Afin de modéliser la topologie du réseau et les interactions entre les différents routeurs, nous avons développé la Méthode de Transition Graphique (MTG). Basée sur l'approche de représentation graphique, elle permet de fournir une modélisation générale du réseau ainsi que le plan de routage explicite adopté et qui sera utilisé dans la conception du système d'équations différentielles. Cela se fait en représentant la topologie du réseau et ses composants (routeurs, liens, chemins, etc) par des matrices d'incidence de 0 et 1. A l'issue de cette seconde étape, un modèle dynamique

d'état a été construit. Il est capable d'estimer l'état de charge du réseau et de calculer les taux de transfert de flux sur les chemins explicites correspondant au plan du routage adopté. L'intérêt de ce modèle réside dans sa capacité à fournir une représentation générale et sans complexité concernant non seulement le réseau mais aussi dans la mesure des flux de trafic dans un domaine IP/MPLS. Notons que cette représentation peut être facilement étendue à d'autres types de réseau.

La troisième contribution consiste à exploiter le modèle dynamique d'état proposé, ainsi que les résultats expérimentaux lors du calcul des flux de trafic transférés afin d'éviter toute situation de saturation ou de congestion des chemins du réseau. Le but est d'améliorer les précisions des mesures du plan de routage effectué tout en surveillant et contrôlant la bande passante résiduelle des chemins utilisés. En s'appuyant sur le modèle dynamique d'état, nous avons proposé deux modèles de contrôle de trafic sur l'entrée du réseau. Les deux modèles sont basés sur l'information sur la bande passante résiduelle des chemins du réseau. En mesurant ce dernier paramètre, nous agissons de façon à minimiser le trafic sur l'entrée dans le premier modèle, et dans le second à réorienter le trafic vers d'autres chemins non congestionnés ou non utilisés. Dans le premier cas, nous proposons un modèle de contrôle en rétroaction sur l'entrée aux chemins du réseau, où le trafic peut être momentanément ralentis et modulés à des valeurs inférieures à la demande. Ce modèle se compose d'un système d'équations différentielles qui exploite les informations sur les bandes passantes résiduelles des chemins concernés. Ensuite, il génère des variables de contrôle sur l'entrée de ces chemins, de façon à minimiser le trafic d'entrée en fonction de la capacité résiduelle des chemins. Dans le deuxième cas, nous proposons un modèle original, basé sur les méthodes précédentes de représentation. Ce modèle adopte un mécanisme d'adaptation dynamique de

routage afin de garder les chemins qui offrent la meilleure QdS. Le principe de ce modèle consiste à recueillir les informations sur la topologie et le plan de routage dans le réseau. Ensuite, en surveillant la bande passante résiduelle de chaque chemin utilisé, il agit de façon à réorienter le trafic vers un autre chemin non utilisé, si le chemin surveillé présente une situation de saturation ou de congestion. L'étude de ces deux techniques a permis de présenter le problème de contrôle de congestion selon différentes manières, en proposant à chaque fois un modèle original, simple et efficace.

Structure de la thèse

Ce mémoire se compose de deux parties, la première contient les chapitres 1 et 2. Les chapitres 3 et 4 forment la seconde partie.

Dans le premier chapitre nous décrivons les différentes techniques, technologies et protocoles utilisés actuellement dans les réseaux IP pour garantir la QdS des paquets et des flux. Nous commençons par illustrer les techniques de l'ingénierie de trafic dans les réseaux IP ainsi que les réseaux IP/MPLS, avant d'expliquer le protocole MPLS, capable de contrôler et de gérer la QdS, ainsi qu'il définit différentes options d'ingénierie de trafic. Ensuite, nous continuons la description des mécanismes de qualité de service, notamment IntServ et DiffServ. Cette dernière norme qui est proposée par l'IETF (Internet Engineering Task Force) permet une différenciation du traitement reçu au niveau de chaque routeur. Enfin nous présentons les techniques qui associent MPLS, l'ingénierie de trafic et DiffServ, avant de détailler la description de deux modèles pour l'ingénierie de trafic par équilibrage de charge dans les réseaux IP/MPS développés au sein du laboratoire LAGIS.

Dans le chapitre 2, on introduit une méthode pour l'évaluation des per-

formances par simulation de ces deux modèles. Cette étude est réalisée sur une plateforme de test utilisant le simulateur de réseaux NS-2. Après une analyse des différents critères d'évaluation, nous présentons un état de l'art ainsi qu'une comparaison des générateurs de topologies existants. Ensuite, différents tests des simulations seront réalisés pour analyser la capacité de passage à l'échelle et la réponse aux changements des différents paramètres de trafic et de topologie des deux modèles.

Suite aux résultats recueillis lors du chapitre 2, la connaissance de l'état de charge du réseau apparaît indispensable pour améliorer les performances de tels modèles. La conclusion sur l'état de charge du réseau, nous mènera à proposer dans le chapitre 3 un modèle dynamique d'état basé sur la théorie différentielle de trafic. Nous développons deux méthodes de représentation basées sur deux approches à compartiments et graphique, afin de proposer une modélisation générale du réseau capable de mesurer dynamiquement son état de charge et calculer les flux de transfert du trafic. Ensuite nous appliquons ce modèle sur un réseau de type IP/MPLS implémentant des techniques de l'ingénierie de trafic, et notamment le routage explicite. Ainsi, nous introduisons un modèle non linéaire d'état basé sur le modèle proposé caractérisant les phénomènes avec et sans retard.

Dans le Chapitre 4, nous proposons deux méthodes d'adaptation dynamique des chemins du réseau en fonction de la capacité de leurs bandes passantes résiduelles. En appliquant le modèle proposé dans le chapitre 3, nous mettons en oeuvre deux modèles de contrôle de congestion sur les réseaux de type IP/MPLS. Le premier modèle est basé sur un contrôle en rétraction sur l'entrée des chemins du réseau, tandis que le deuxième consiste à réorienter le trafic au cas où le chemin est saturé.

Une série de conclusions et perspectives clôturant ce travail.

Chapitre 1

Les réseaux IP/MPLS et l'ingénierie de trafic avec des contraintes de QdS

1.1 Formulation de la technique de l'ingénierie de trafic dans le contexte des réseaux IP/MPLS

1.1.1 L'évolution de IP à MPLS

La croissance exponentielle des besoins en termes de réseaux implique, en plus d'une augmentation considérable des ressources, l'intégration de nouvelles fonctionnalités. Celles-ci sont nécessaires, d'une part pour utiliser de manière efficace les ressources ainsi disponibles, et d'autre part pour satisfaire les récents besoins des nouvelles applications (Fig.1.1)[81]. Cependant, l'intégration de ces fonctionnalités dans l'internet nécessite l'utilisation de

techniques de plus en plus performantes. Ce qui a amené les Fournisseurs d'Accès Internet (FAI) à introduire de nouvelles technologies dans le coeur de réseau tel que l'IP (Internet Protocol) [51], l'ATM (Asynchronous Transfer Mode) [68], le PoS (Packet over Sonet/SDH), ou encore le MPLS (Multi Protocol Label Switching) [85] (Fig.1.2).

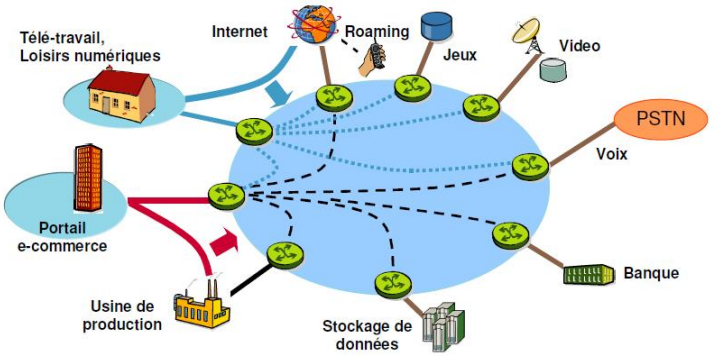


FIG. 1.1 – *Evolution des services dans le réseau futur*

L'évolution de IP vers MPLS est née de la confrontation des différentes solutions élaborées pour mieux intégrer l'ATM [68] et l'IP [51]. Pour ce qui est de la comparaison avec des technologies issues de l'ATM, l'essentiel des problèmes réside dans le choix des protocoles de recherche de chemins, et dans la différence d'approche mode connecté/non connecté, aussi dans la complexité à assembler/segmenter des paquets de longueur variable en cellules de longueur courte à très haut débit (difficulté pour l'ATM à supporter des interfaces 2.5 Gbit/s et au-delà) [22].

L'idée a consisté de ne garder de l'ATM que sa capacité de transfert et remplacer les protocoles de commandes jusque là associés (Q.2931[80], PNNI[2]) par un contrôle direct des matrices de commutation par les protocoles de routage utilisés dans les réseaux IP (OSPF [71], ISIS [15]) [22]. Les motivations pour l'élaboration d'une telle technique vont bien au delà de la facilité d'intégration de l'IP et l'ATM. Il s'agit en effet d'enrichir les

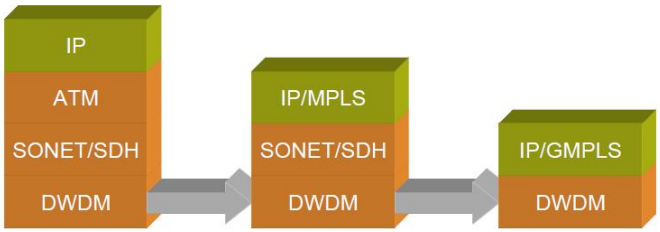


FIG. 1.2 – *Evolution des réseaux coeurs*

capacités de la technologie IP, également:

- La capacité à mettre en oeuvre une ingénierie de trafic et par là même permettre la définition de contrats de qualité de service.

- La capacité à isoler des trafics autorisant ainsi la création de réseaux privés virtuels (VPN [84]) directement à partir du réseau et non plus seulement sur la base de techniques de tunnels. Il s'agit ici d'intégrer dans le service IP les vertus des technologies relais de trame ou ATM qui travaillent en mode connecté.

- La capacité à croître: un élément fort de l'architecture MPLS est de permettre les encapsulations successives de labels. Il s'agit en fait d'une extension non limitée du concept établi dans l'ATM avec l'empilement VP, VC.

1.1.2 Définition de l'ingénierie de trafic

Afin de contrôler la croissance des débits d'accès et la convergence des services dits "triple play" (voix/visioconférence, télévision/vidéo à la demande), tout en réduisant les coûts d'investissement, deux types d'ingénierie se distinguent: l'ingénierie de réseau et l'ingénierie de trafic.

L'ingénierie de réseau consiste à adapter la topologie du réseau au trafic. En se basant sur des prédictions sur le trafic, elle installe ou modifie la topologie du réseau (ajout de liens, routeurs, etc..). L'ingénierie de réseau

s'effectue sur une longue échelle de temps (mois/année), le temps nécessaire pour l'installation de nouveaux équipements ou circuits. Les prédictions, sur lesquelles se base l'ingénierie de réseau, ne correspondent jamais entièrement à la réalité. Dans certains cas, un changement de trafic se produit dépassant les prédictions. La modification de la topologie du réseau pour s'adapter à ce changement peut ne pas être suffisamment rapide.

Par ailleurs, l'ingénierie de trafic [4] consiste à adapter le trafic à la topologie du réseau. L'expression "Ingénierie de trafic" désigne l'ensemble des mécanismes de contrôle de l'acheminement du trafic dans le réseau afin d'optimiser l'utilisation des ressources et de limiter les risques de congestion (Fig. 1.3). L'objectif de l'ingénierie de trafic est de maximiser la quantité de trafic pouvant transiter dans le réseau, tout en maintenant la qualité de service (bande passante, délai ...) offerte aux différents flux de données. En général, le processus d'ingénierie de trafic est divisé en quatre phases qui sont répétées de manière itérative [4]:

- La définition de politiques appropriées de contrôle qui gouvernent le réseau,
- La mise en application de mécanismes de rétroaction pour l'acquisition des données de mesure du réseau opérationnel,
- L'analyse de l'état du réseau,
- La mise en application de l'optimisation du réseau.

Par ailleurs, l'ingénierie de trafic (trafic engineering: TE) est donc nécessaire dans l'Internet. En effet, les protocoles de routage interne (IGP, Interior Gateway Protocol) utilisent toujours le plus court chemin pour expédier le trafic ([32], [83]). Malgré le fait que l'approche du plus court chemin conserve les ressources du réseau et bien qu'elle soit très simple à appliquer aux grands réseaux, elle ne fait pas toujours la meilleure utilisation de ces ressources et peut également introduire les problèmes suivants:

- Les plus courts chemins des différentes sources peuvent se superposer sur certains liens provoquant ainsi la congestion de ces liens,
- Le trafic d'une source à un destinataire peut dépasser la capacité d'un lien alors qu'un autre chemin légèrement plus long reste sous-utilisé.

En plus, ces propositions souffrent de plusieurs limitations [28]:
- Les contraintes sur le trafic (par exemple, éviter certains liens pour un trafic particulier d'une source à une destination) ne peuvent pas être gérées,
- Les modifications des métriques des liens pour permettre l'association explicite du trafic à un chemin tend à avoir des effets difficilement contrôlables sur le reste du réseau,
- Le partage de charge ne peut pas être effectué entre les chemins de coûts différents,

L'utilisation de MPLS, de sa capacité d'acheminer le trafic à travers des chemins explicites et de sa flexibilité de gestion du trafic qu'il apporte, permet d'éviter ces limitations.

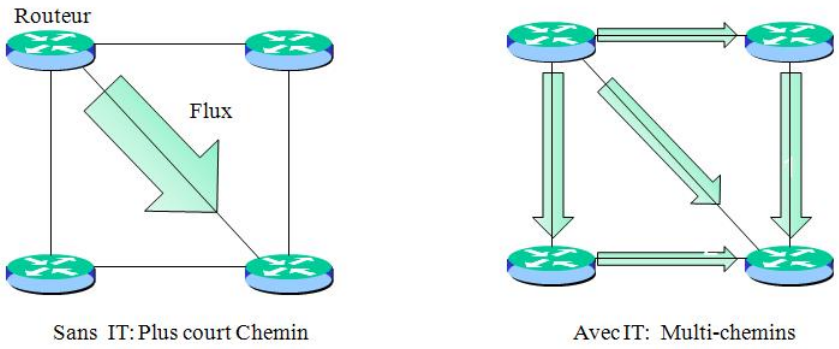


FIG. 1.3 – *Exemple d'utilisation du concept de l'ingénierie de trafic IT*

1.1.3 Ingénierie de trafic basée sur IP: IP-TE

1.1.3.1 Introduction

Les protocoles Internet constituent la famille la plus populaire de protocoles des systèmes ouverts non propriétaires, car ils peuvent être utilisés pour communiquer entre n'importe quel ensemble de réseaux interconnectés et sont aussi bien adaptés aux communications dans les réseaux locaux que dans les réseaux à grande échelle. Les plus connus sont IP (Internet Protocol) [51] et TCP (Transmission Control Protocol) [49] [88]. Les protocoles IP ont été développés dans les années 70, TCP/IP fut introduit dans Unix BSD (Berkeley Software Distribution) et amena la brique de base de ce qu'est actuellement l'Internet et le "Web". Les documentations et normalisations de l'Internet sont consignées dans des RFC (Request For Comments).

IP [51] est un protocole qui permet l'adressage des machines et le routage des paquets de données. Il correspond à la couche 3 (réseau) de la hiérarchie des couches ISO. Son rôle est d'établir des communications sans connexion de bout en bout entre des réseaux, de délivrer des trames de données (datagrammes) "best effort", et de réaliser la fragmentation des datagrammes pour supporter des liaisons n'ayant pas la même MTU (Maximum Transmission Unit, c'est-à-dire la taille maximale d'un paquet de données).

1.1.3.2 Routage dans le réseau IP

Le routage est l'une des principales fonctionnalités de la couche réseau qui a la responsabilité de décider sur quelle interface de sortie, un paquet entrant doit être retransmis. D'une manière générale, on distingue la remise directe, qui correspond au transfert d'un datagramme entre deux ordinateurs du même réseau, et la remise indirecte qui est mise en oeuvre quant au moins

un routeur sépare l'expéditeur initial et le destinataire final. En effet, le routage IP est un routage de proche en proche. Dans un routeur, une table de routage IP contient la liste des destinations connues localement afin d'associer une interface de sortie à toutes les destinations (next-hop). Mais, avec la fantastique expansion de l'internet, l'explosion des tables de routage fut l'une des raisons pour lesquelles la décision de structurer l'Internet en des ensembles de réseaux chacun soumis à une administration commune. Pour cela, les routeurs dans l'Internet sont organisés de manière hiérarchique. Des routeurs sont utilisés pour envoyer les informations dans des groupes de routeurs particuliers sous la même responsabilité administrative. Ces groupes sont appelés des Systèmes Autonomes (Autonomous Systems: AS). Ces routeurs permettent à différents Systèmes Autonomes de communiquer entre eux. Ils utilisent des protocoles de routage appelés Exterior Gateway Protocols (EGP) tels que (EGP [69] ou BGP [61]). D'autres routeurs appelés Interior Gateway Router, utilisent des protocoles de routage appelés Interior Gateway Protocols (IGP) tels que (RIP [63] [45], OSPF [71], IGRP(Interior Gateway Routing Protocol)[20], IS-IS [15])) (Fig.1.4). En focalisant plus particulièrement sur les protocoles intra-domaines, on peut distinguer les protocoles de routages classiques qui cherchent à calculer le plus court chemin entre une source/destination, et on peut les répartir entre deux grandes familles: les protocoles à vecteur de distance (RIP: Routing Information Protocol [63] [45]) et les protocoles à états de liens(OSPF: Open Shortest Path First [71]). On trouve aussi les approches de routage avec qualité de service (QdS), où le routage doit alors prendre en compte l'état réel des liens et des routeurs afin d'effectuer un routage par "qualité de service" (QOSPF [23], SW [41], Q-Routing [10]).

Type de protocole	Type de routage	TCP/IP	OSI	Cisco
IGP	Vecteur Distance	RIP Routing Information Protocol		IGRP Interior Gateway Routing Protocol
	Etat de Liens	OSPF Open Shortest Path First	ISIS Intermediate System to Intermediate System	//////
	Hybride	//////		EIGRP Ehanced Interior Gateway Routing Protocol
EGP		BGP	IDRP	//////

FIG. 1.4 – *Exemples de protocoles de routage dans Internet*

1.1.3.3 Ingénierie de trafic dans les réseaux IP

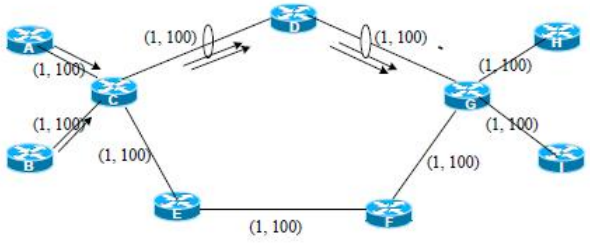
Il existe plusieurs méthodes d'ingénierie de trafic dans les réseaux IP (IP-TE). Une solution consiste à manipuler les métriques des protocoles de routage IP. En effet, le routage classique IP repose sur le plus court chemin vers une destination donnée. Tout le trafic vers une même destination emprunte le même chemin. Il arrive que le chemin IP soit congestionné alors que des chemins alternatifs sont sous-utilisés. L'ingénierie de trafic avec IP (IP-TE) représente une solution pour dépasser les limitations du routage IP. Elle calcule un ensemble de chemins pour répondre aux demandes de trafic sans saturer les liens, et calcule des métriques pour satisfaire ces chemins. Ensuite, un partage de charge offert par le protocole de routage peut être utilisé pour permettre à un routeur de partager équitablement la charge entre tous les chemins de coût égal. Un routage optimal (selon un critère d'optimisation donné) nécessite la détermination des coûts sur chaque lien qui répondent au critère d'optimisation. Un large ensemble de solutions a été proposé pour l'IP-TE [30] [31]. Toutes ces solutions consistent à optimiser les poids des liens utilisés (débits, délais, ..) par la suite, et qui peuvent changer selon le protocole de routage appliqué. Cette ingénierie de trafic basée sur

l'optimisation des métriques IP peut bien fonctionner uniquement sur de petites topologies avec un faible nombre de routeurs d'accès. Changer les coûts des liens sur tous les chemins pour une grande topologie reste très difficile à mettre en oeuvre en tenant compte des risques d'instabilité, des problèmes de convergence IGP et des problèmes liés aux boucles de routage. Afin de gérer l'aspect dynamique du réseau par exemple, des solutions proposés dans [78] prennent en compte les scénarios de cas de pannes dans le réseau. Dans [33], les auteurs ont considéré des cas de pannes qui peuvent se produire également lors du changement de trafic.

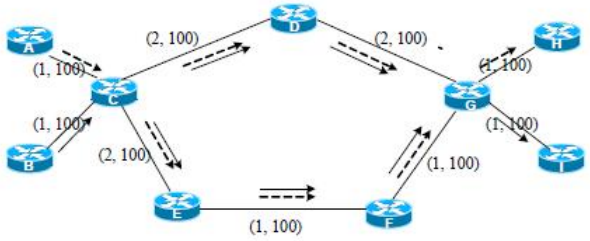
Les solutions d'IP-TE ne s'appliquent généralement pas lorsque l'on a des chemins de capacités différentes. En effet, les routeurs ne sont pas capables de faire un partage de charge tenant compte de la capacité des liens.

La Fig.1.5 illustre une application de routage IP, IP-TE [18]. Elle représente un réseau comportant 9 noeuds et 9 liens bidirectionnels. Les liens sont caractérisés par leurs métriques IP (égales à 1) et leurs capacités en Mbit/s (égales à 100 Mbit/s). Deux demandes de trafic arrivent au réseau de 70 Mbit/s. Dans le premier cas correspondant au routage IP, les deux trafics empruntent le même plus court chemin avec un coût de 2. Ce même chemin est de capacité 100 Mbit/s, qui est soumis à une charge de 140 Mbit/s. Par conséquent, la congestion inévitable va entraîner une perte de paquets et ainsi une dégradation de la qualité de service. Une solution pour remédier à ce problème est d'utiliser le routage IP-TE avec un partage de charge donné par le protocole de routage IGP. Lorsqu'il y a plusieurs choix de courts chemins de même coût pour aller à une destination donnée, un routeur peut partager équitablement la charge sur ces chemins. Il envoie alors la même quantité de trafic sur tous les chemins. Ce mécanisme est appelé ECMP [27]. La Fig. 1.5 montre que si on applique le routage IP-TE, en mettant les métriques à 2 sur

les liens C-D, D-G et C-E, on se retrouve alors avec deux chemins de coût égal (coût de 4) entre C et G. Dans ce cas, le routeur G effectue un partage de charge entre les deux chemins. Il envoie 50% du trafic sur le chemin du haut et 50% du trafic sur le chemin du bas. On a donc une charge de 70 Mbit/s sur les deux chemins. la congestion du réseau est alors évitée.



Cas de congestion avec le routage IP



Routage IP-TE avec partage de charge

FIG. 1.5 – *Ingénierie de trafic basée sur IP*

1.1.4 Ingénierie de trafic basée sur MPLS

1.1.4.1 Introduction

La convergence des réseaux multiservices, qui transportent le trafic Internet, V2oIP (Voice/Video over IP), IP-TV, vidéo à la demande et le trafic VPN, nécessite une optimisation de l'utilisation des ressources pour limiter les coûts d'investissement, une garantie stricte de la qualité de service (QdS) et une disponibilité élevée. A tous ces besoins s'ajoute celui de limi-

ter les coûts d'exploitation. Des mécanismes d'ingénierie de trafic s'avèrent alors nécessaires pour répondre à tous ces besoins. De nombreuses méthodes d'ingénierie de trafic pour les réseaux IP ont été proposées. L'utilisation de l'architecture MPLS (MultiProtocol Label Switching) est l'une d'elles, car elle s'avère bien adaptée aux objectifs d'ingénierie de trafic. En effet, MPLS offre le routage explicite, permettant la création de chemins routés de façon explicite, indépendamment de la route IP (qui repose sur un plus court chemin vers la destination). On appelle ces chemins tunnels MPLS ou Label Switched Path (LSP) MPLS. Ainsi, ce routage explicite assure une bonne souplesse pour optimiser l'utilisation des ressources et assurer un bon partage de charge. L'application de MPLS à l'ingénierie de trafic est appelée MPLS-Traffic Engineering (MPLS-TE). La technologie MPLS-TE permet un routage explicite par contrainte, elle inclut des protocoles de découverte de la topologie, de calcul de chemins explicites et d'établissement de ces chemins.

1.1.4.2 Technologie MPLS

Avec l'augmentation du débit des liens de communication et du nombre de routeur dans l'Internet, l'IETF [48] a définit le nouveau protocole MPLS (Multi Protocol Label Switching: commutation d'étiquettes adaptée au contexte multi-protocole)[85] pour faire face au goulot d'étranglement induit par la complexité de la fonction de routage.

La technologie MPLS (Multiprotocol Label Switching)[85] consiste à aiguiller des paquets non plus en fonction de leur adresse IP de destination, mais en fonction d'un label inséré entre l'entête de niveau 2 et l'entête IP. Ainsi, on dit souvent de MPLS qu'il est un protocole de niveau 2.5 (Fig.1.6).

Le protocole MPLS basé sur le paradigme de la commutation de label dé-

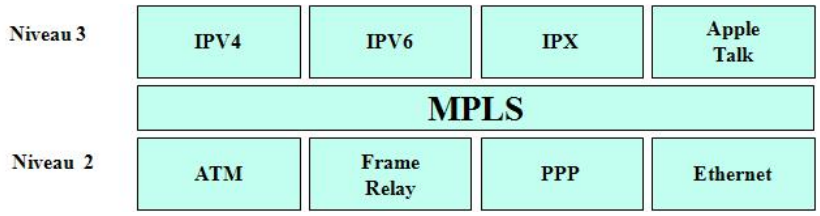


FIG. 1.6 – *Positionnement de MPLS par rapport au modèle OSI*

rive directement de l'expérience acquise avec ATM [68] (étiquettes VPI/VCI: Virtual Path/Channel Identifier). Ce mécanisme est aussi similaire à celui de Frame Relay [12] ou des liaisons PPP (PPP: Point to Point Protocol) [86]. L'idée de MPLS est d'ajouter un label de niveau liaison point-à-point (niveau 2 dans la hiérarchie ISO) aux paquets IP dans les routeurs d'entrée du réseau d'un fournisseur d'accès internet (FAI) ou d'un système autonome. Le réseau MPLS va pouvoir créer des LSP (Label Switching Path: Chemin à commutation de label), similaires aux VPC (Virtual Path Connection : Chemin de connexion virtuelle) d'ATM [68]. Ces LSP définissent une route de bout en bout traversant de multiples routeurs internes au domaine pour relier un de ses routeurs d'entrée à un de ses routeurs de sortie. En pratique chaque label constitue une adresse locale caractérisant la connexion, et un LSP correspond à une cascade d'adresses locales ou labels entre un routeur d'entrée et un routeur de sortie. Les LSP sont des chemins de LER (Label Edge Router : Router à label de bordure) à LER positionnés par l'opérateur. La construction de ces chemins se fait dans les LSR (Label Switching Routers : Routeurs à commutation de label), qui insèrent leur identification dans des messages de contrôle du protocole de signalisation comme LDP (Label Distribution Protocol: Protocole de distribution de labels)[1] pour le routage implicite, et qui est été étendu dans le but d'établir des LSP explicites ER-LSP (Explicitly Routed Path) en CR-LDP (Constraint-Based Routing LDP)[52] (Fig.1.7) ou

RSVP-TE (protocole de reservation des ressources)[11] (Fig.1.8).

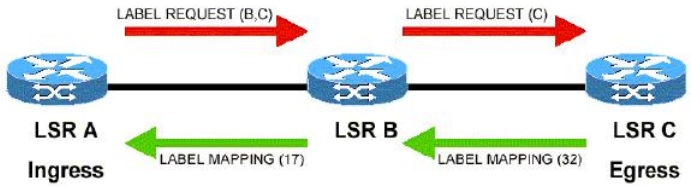


FIG. 1.7 – *Etablissement d'un LSP par CR-LDP*

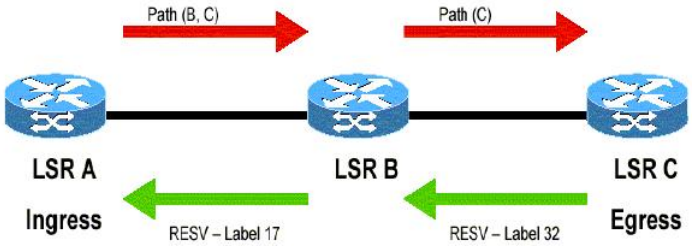


FIG. 1.8 – *Etablissement d'un LSP par RSVP-TE*

Lorsqu'un paquet entre dans le coeur du réseau, une entête MPLS lui est ajouté (figure 1.9), qui détermine le LSP par lequel il sera acheminé.

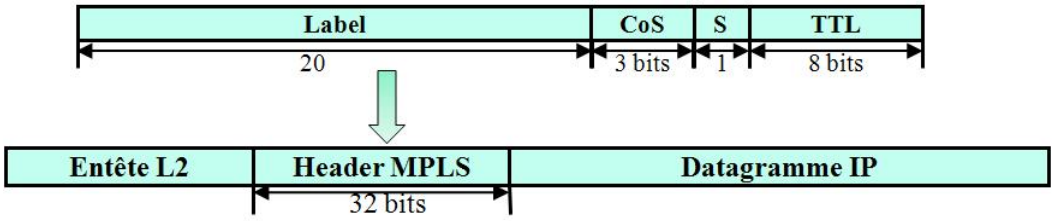


FIG. 1.9 – *Entête MPLS*

Afin de déterminer cette entête, la FEC (Forwarding Equivalent Class: classe d'équivalence pour le routage) et le type de service du paquet jouent un rôle déterminant. Une FEC qui peut être une liste d'adresses IP, où chaque adresse correspondant à une machine particulière ou une adresse de réseau. Cette spécification permet de regrouper par destination les flux d'un même

LSP. Chaque LSP a donc pour caractéristique un ensemble de FEC, et un ensemble de classes de services. Une table de correspondances dans chaque routeur de bordure (LER) permet d'affecter chaque flux (en fonction de sa classe et de sa destination) à un LSP. Une fois le paquet encapsulé, il sera routé dans son LSP via les routeurs du coeur du réseau (LSR) grâce à leur table de commutation. Une fois arrivé dans le LSR de sortie, l'entête MPLS est retiré du paquet, et celui-ci est relâché dans le réseau IP pour continuer sa route. Les LSR d'entrée sont appelés Ingress, et les LSR de sortie Egress (Fig.1.10)[17].

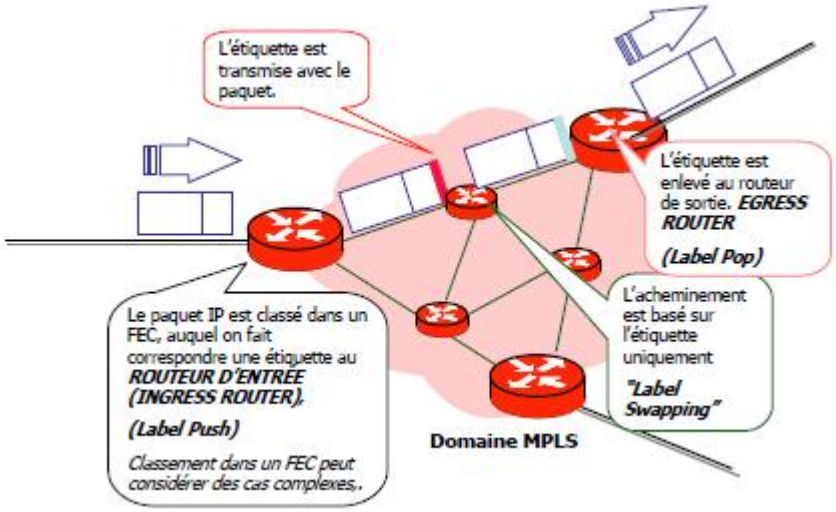


FIG. 1.10 – *Commutation des paquets dans MPLS*

L'établissement des LSP permet en particulier de passer en mode connecté dans les réseaux IP et d'effectuer un routage explicite indépendant de la route IP. Dans un domaine MPLS, tout le mécanisme de commutation de labels et d'établissement des LSP tourne sur des routeurs IP qui disposent en plus des fonctionnalités MPLS. En effet, l'un des avantages de MPLS est l'accélération de la commutation des paquets au sein d'un backbone IP. Les grandes innovations par rapport au routage IP traditionnel (IGP, EGP[69]),

sont:
- La manipulation de tables de routage de bien plus petite taille et des champs de petite taille pour les labels conduisant ainsi à des temps de commutation extrêmement rapides (traitement matériel plutôt que logiciel),
- La détermination des LSP (les chemins) est basée sur d'autres critères que l'adresse de destination uniquement (classe de service, groupes d'adresses de destination),
- Un ensemble de règles et de paramètres permettant de calculer un routage "optimal" (autre que le plus court chemin) pour l'ingénierie du trafic et la qualité de service,
- Des mécanismes de reroutage en cas de panne avec la possibilité de réserver des LSP de secours pour certains trafics.

1.1.4.3 MPLS-TE

L'apport de l'ingénierie de trafic est de permettre à l'administrateur du réseau la distribution des multiples flux de trafic sur les différentes liaisons existantes afin de maîtriser la QdS dans le réseau. Il devra donc:
- Maximiser l'utilisation des noeuds et des liens du réseau,
- Répartir le trafic dans le réseau pour éviter les surcharges et minimiser l'impact des pannes,
- Prévoir une capacité réseau en cas de panne ou de surcharge ponctuelle,

L'ingénierie de trafic MPLS permet en plus: [52]
- Le routage explicite,
- La spécification de la QdS des LSP créés,
- La spécification des paramètres des trafics entrants,
- L'association des trafics aux différents LSP déjà existants,
- La création et le maintien de routes de secours.

MPLS-TE (MPLS - Traffic Engineering) est basé sur le concept de routage de tunnels (traffic trunk). Le tunnel est unidirectionnel et est défini par deux LER (Ingress et Egress) (Fig.1.11)[18]. Le chemin emprunté par le tunnel est un LSP. un LSP peut contenir un ou plusieurs tunnels. Plusieurs LSP peuvent exister entre deux LER, et emprunter des chemins différents. Cela permet d'une part d'offrir des routes différentes correspondantes à la qualité de service requise pour les flux transportés grâce à la répartition très fine qui peut être faite, et d'autre part de créer des LSP de secours en cas de panne ou de surcharge sur les LSP initiaux. La création des LSP se fait en plusieurs étapes.

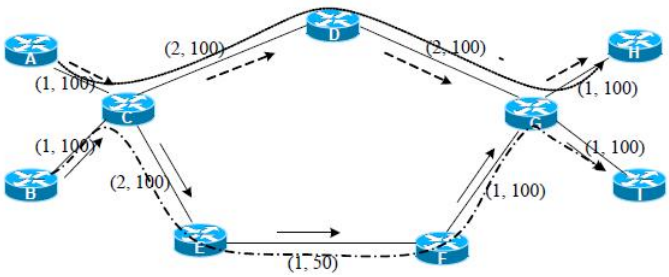


FIG. 1.11 – *Exemple d'un routage MPLS-TE*

Une fois les tunnels déterminés (groupes de flux ayant la même FEC et le même couple Ingress-Egress), puis une fois les tunnels regroupés en un ou plusieurs LSP selon le même couple Ingress-Egress, il faut créer les routes dans le coeur du réseau pour propager ces LSP. Les LSP sont routés de façon explicite en tenant compte des contraintes de trafic (bande passante, etc.) et les ressources disponibles dans le réseau. Ces LSP vont être utilisés par la suite pour transporter du trafic entre les routeurs d'accès du réseau. MPLS-TE combine le routage explicite offert par MPLS et le routage par contrainte.

En effet, le routage par contrainte repose sur une fonction de découverte

dynamique de la bande passante réservable sur un lien, une fonction de calcul de chemin explicite contraint, ainsi qu'une fonction d'établissement de LSP explicites avec réservation de ressources et distribution de labels le long du chemin explicite. Ce mécanisme repose sur trois fonctions principales:

- La fonction de découverte de la topologie: elle permet à tous les routeurs d'avoir une vision actualisée de la topologie TE et en particulier de la bande passante résiduelle réservable sur les liens. Cette fonction est assurée par un protocole IGP-TE. La topologie TE est enregistrée par chaque routeur du réseau dans une base de données TE appelée TED (pour TE Database), qui enregistre pour chaque lien du réseau les paramètres TE comme: la bande passante maximale, la bande passante maximale réservable, la bande passante disponible et d'autres métriques.
- La fonction de calcul de chemins: elle permet de calculer les chemins pour les LSP en utilisant un algorithme de routage par contrainte. Elle prend en entrée les contraintes des LSP (bande passante, groupes à inclure/exclure, etc.) et la topologie TE alimentée par le protocole IGP-TE. Deux approches complémentaires seront détaillées dans le paragraphe suivant et qui appliquent l'ingénierie de trafic par équilibrage de charge. Les figures 1.12 et 1.13 [81] montrent deux exemples pour l'établissement d'un LSP.
- La fonction de signalisation des LSP: une fois la route explicite pour un LSP calculée, la fonction de signalisation intervient pour établir le LSP. Cet établissement comprend le routage explicite du LSP le long de la route explicite, la réservation de ressource sur les liens traversés ainsi que la distribution des labels sur le chemin. Elle est réalisée par des protocoles de signalisation comme RSVP-TE [3] ou CR-LDP [52].

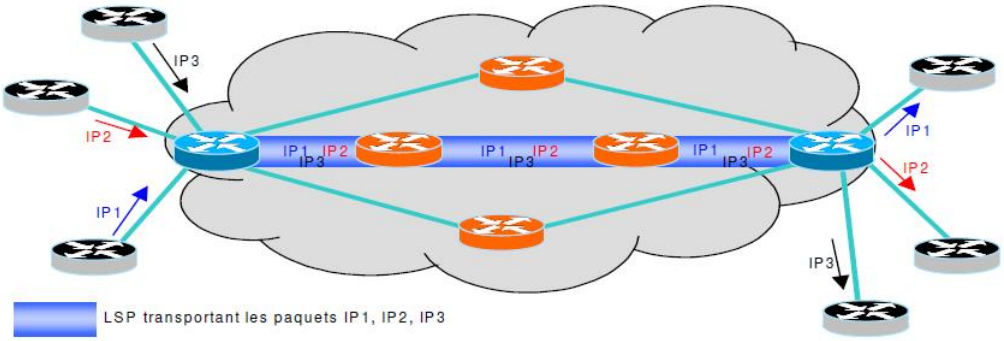


FIG. 1.12 – *Un LSP pour toutes les classes de trafic*

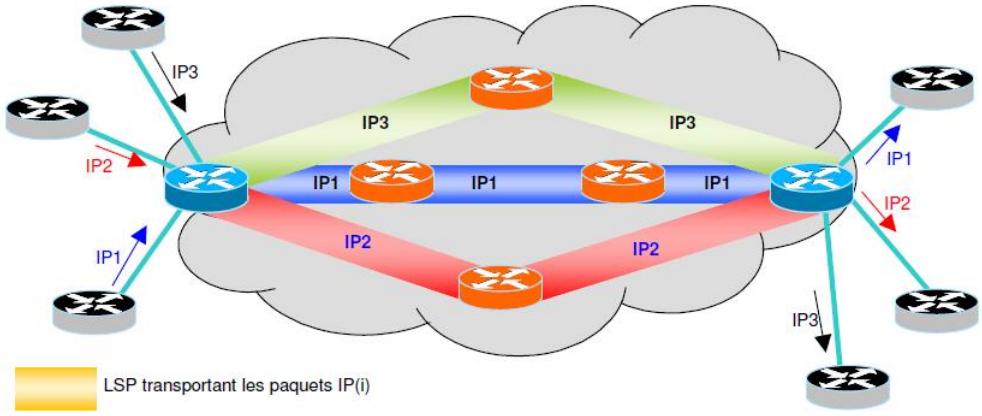


FIG. 1.13 – *Un LSP par Source/Destination et par classe*

1.1.4.4 Approche complémentaire: l'équilibrage de charge par routage multi-chemins

L'ingénierie de trafic consiste à réaliser l'équilibrage de charge sur le réseau de façon à optimiser les performances. Cet équilibrage de charge est réalisé en mettant en oeuvre des protocoles et des algorithmes de façon à pré-calculer et à réserver un ensemble de chemins afin de répartir de façon optimale le trafic sur les différentes routes. L'équilibrage de charge permet donc de mieux distribuer le trafic sur le réseau. Dans ce paragraphe, on présente une technique de l'équilibrage de charge, à savoir le routage multi-

chemins. Les algorithmes de routage classiques utilisent une route unique pour acheminer les paquets de la source à la destination. Pour utiliser au mieux les ressources du réseau, il peut être intéressant d'effectuer un routage multi-chemins. En effet, les algorithmes de routage peuvent trouver plusieurs chemins ayant le même coût minimal ou des coûts faibles. En plus, l'utilisation d'une métrique unique peut s'avérer insuffisante pour satisfaire les besoins de certaines applications. Par exemple, une application de visioconférence peut requérir d'une part, une certaine bande passante afin que la qualité de la retransmission vidéo soit correcte et d'autre part, un délai de bout en bout borné pour que les conversations ne soient pas entrecoupées de "blancs" désagréables pour les utilisateurs. Une possibilité est d'utiliser une métrique mixte, composée de plusieurs métriques primitives (telles que la bande passante, le délai, la gigue, etc.). Les algorithmes de routage multi-chemins sont schématiquement composés de deux étapes (Fig.1.14): le calcul de l'ensemble des chemins candidats et la répartition du trafic sur ces chemins.

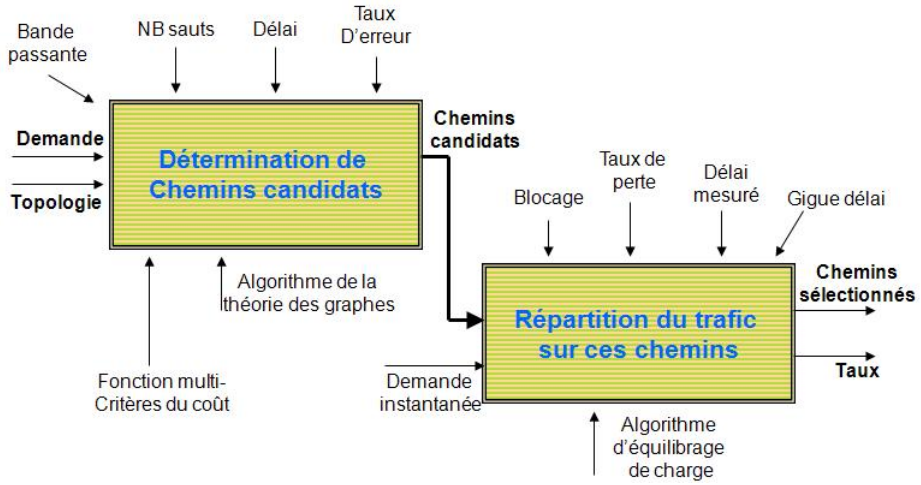


FIG. 1.14 – *Shéma général du routage multi-chemins*

L'ensemble des chemins candidats est un sous ensemble de tous les che-

mins entre une paire source/destination qui vérifient certains critères de qualité de service (QdS) comme la bande passante, le nombre de sauts, le délai... Ces critères, généralement statiques, caractérisent les connexions entre routeurs. En effet, il est parfois préférable de considérer des métriques combinées pour savoir si un chemin dispose des capacités à accueillir un flux avec de multiples contraintes comme par exemple en requérant telle bande passante ou telle contrainte de délai. La valeur de certaines métriques combinées est en fait une heuristique rapportant globalement l'état d'un chemin. Un modèle comme WDP [12] est un algorithme de sélection des chemins candidats basé essentiellement sur les critères bande passante et nombre de sauts.

La deuxième étape consiste à répartir le trafic sur ces chemins candidats. Le taux de répartition du trafic sur chaque chemin candidat dépend d'une métrique basée sur plusieurs critères dynamiques comme le taux de perte de paquets, le délai mesuré, la variation du délai, etc. Dans les différents algorithmes, on cherche à chaque instant à sélectionner le bon nombre de chemins candidats afin de ne pas réduire l'efficacité de la répartition. En effet, un nombre élevé de chemins peut être pénalisant. De manière générale, si à un instant donné, il est impossible d'affecter une nouvelle demande, les modèles remettent alors en cause le plan de routage courant établi par l'étape de sélection des chemins candidats.

Dans [56], des études ont été faites pour comparer plusieurs algorithmes d'équilibrage de charge par routage multi-chemins comme WDP (Widest Disjoint Path) [74], MATE (MPLS Adaptative Traffic Engineering) [28] et LDM (Load distribution in MPLS network) [87].

1.2 Modèles pour la qualité de service dans Internet

L'Internet doit permettre le déploiement d'applications multimédia ayant des exigences spécifiques en terme de QdS. Certains services comme les services vocaux ont besoin d'un faible délai point à point et d'une faible gigue. D'autres, comme les trafics de données, nécessitent de faibles taux de perte ou d'erreurs sans retransmission avec éventuellement une certaine garantie de bande passante pour le trafic de données transactionnel. Pour pouvoir garantir la QdS des flux transportés, il va donc falloir utiliser des mécanismes permettant de traiter de manière différenciée les différentes catégories de trafic, ainsi que des protocoles de signalisation de la QdS pour pouvoir allouer des ressources en fonction des besoins des applications. La qualité de service (QdS) se décline principalement en quatre paramètres : débit, délai, gigue et perte. L'introduction de la Qualité de Service (QdS) permet à l'Internet de fournir d'autres classes de service que le traditionnel "best effort". Pour cela il faut que le réseau soit capable d'isoler les flots pour leur fournir la QdS requise en leur offrant un traitement spécifique. Aujourd'hui, l'IETF défini deux approches de QdS : Services Intégrés (IntServ) [50] et Services Différenciés (DiffServ) [25].

1.2.1 IntServ

Le modèle IntServ [50] définit une architecture capable de prendre en charge la QdS en définissant des mécanismes de contrôle complémentaires sans toucher au fonctionnement IP. C'est un modèle basé sur le protocole de signalisation RSVP [11]. Dans ce modèle, les routeurs réservent les ressources pour un flot de données spécifiques en mémorisant des informations d'état

(Fig. 1.15). IntServ définit deux nouveaux types de services:
- Guaranteed Service (GS) : garantit la bande passante et un délai d'acheminement limité (Audio, Vidéo) mais pas de gestion de la gigue.
- Controlled Load : est équivalent à un service Best Effort dans un environnement non surchargé.

Une architecture telle que IntServ/RSVP qui requiert le maintien des états par flot pose certains problèmes pour le "passage à l'échelle". Une augmentation très importante du nombre de flots à gérer se traduit par un accroissement de la charge de traitement qui peut devenir insupportable. Cependant, le maintien des états n'est pas le seul problème. Il y a également celui dû à la multiplication des classes qui génèrent des complexités au niveau ordonnancement. L'autre problème aussi est le fait que les fonctions de IntServ sont implémentées dans chaque routeur. Cela peut ralentir les traitements et par exemple entraîner une baisse du débit.

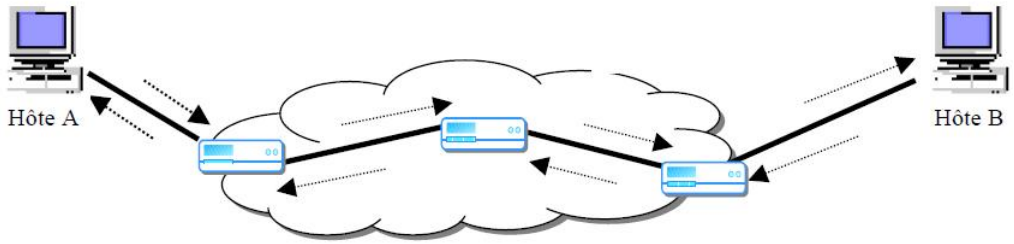


FIG. 1.15 – *Principe général du modèle IntServ*

1.2.2 DiffServ

Le principe de DiffServ [25] (Fig.1.16, Fig.1.17) est de différencier des agrégations de flots regroupés par classes de services (temps réel, transactionnel ou" best effort" par exemple). Cela signifie que DiffServ n'utilise pas de mécanisme de réservation de ressources et offre aux flots une QdS bien

meilleure que celle du "Best effort" (IP normal). L'avantage de ce modèle est d'une part, sa proximité avec IP, ce qui permet une implantation aisée, et d'autre part ses capacités d'évolution face à l'introduction de nouveaux services. Le modèle DiffServ apporte une QdS différenciée pour chaque classe de service selon un contrat prédéfini avec l'émetteur des flux de données. Ce contrat correspond à un ensemble de paramètres (bande passante garantie, pic de données acceptés,...) ainsi que les micro-flux associés à chaque classe de service. Les flux dont les besoins sont similaires, sont regroupés (agrégation), puis en fonction du champ DSCP: Differentiated Services Code Point (marquage de la classe), sont traités spécifiquement par les routeurs pour déterminer le comportement PHB (champ PHB: Per Hop Behavior): choix de file d'attente, algorithme de gestion, ...

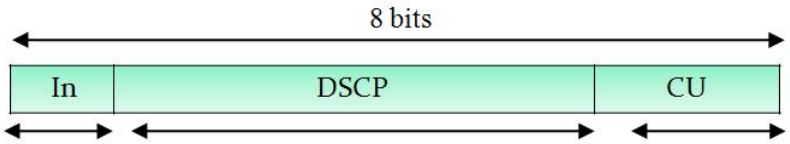


FIG. 1.16 – *Structure du champ DS*

Un domaine DiffServ définit un ensemble de noeuds réseau appliquant une politique de QdS commune. Dans un domaine DiffServ, les champs "TOS" (IPv4) ou "Traffic Class" (IPv6) des entêtes IP sont remplacés par un champ appelé DS dont la structure (Fig.1.16) se compose de:

- Un champ DSCP (DiffServ Code Point) indique la classe du paquet,
- Un champ In indique si le paquet est " In/Out profile " (peut être éliminé ou non),
- Un champ CU n'est pas utilisé pour l'instant.

Dans [75] et [92], trois classes ont été définies dans le cadre de DiffServ:

- Expedited Forwarding (EF) [24] : les flux appartenant à cette classe devront

garantir un délai faible, une gigue réduite et un taux de pertes faible.
- Assured Forwarding (AF) [46] : Quatre classes AF sont définies: AF1, AF2, AF3 et AF4. Chacune de ces classes indépendantes est divisée en trois sous-classes (i.e. AF1 devient AF11, AF12 et AF13).
- Default Behavior (DE) ou Best Effort : les flux appartenant à cette classe n'ont aucune garantie de délai.

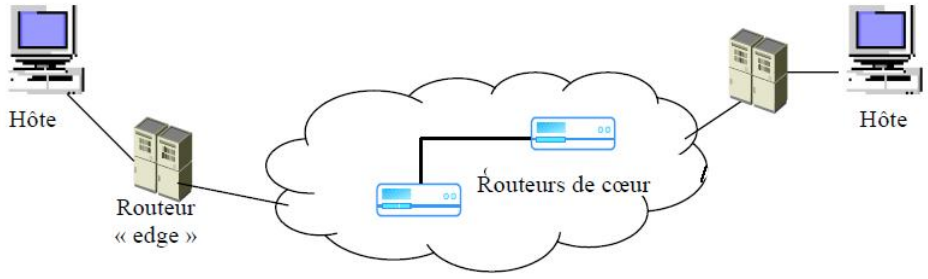


FIG. 1.17 – *Principe général du modèle diffserv*

1.2.3 DiffServ aware MPLS-TE: DS-TE

Avant de formuler les techniques de routage des réseaux supportant le DS-TE [72], nous définissons quelques concepts relatifs au traitement des flux :

- FEC: un groupe de paquets IP qui sont commutés de la même manière dans un domaine MPLS. (e.g. sur le même chemin, avec le même traitement..),
- Behaviour Aggregates (BA): un ensemble de paquets IP passant dans un domaine DiffServ et demandant le même traitement. En d'autres termes, un BA est un ensemble de paquets qui possède le même champ DSCP. Chaque noeud DiffServ détermine, à partir de la valeur du DSCP dans l'entête du paquet, le PHB (Per Hop Behavior) à appliquer (i.e. l'ordonnancement et la

probabilité de perte),
- Ordered Agregate (OA): un ensemble de Behaviour Aggregates liés par un certain partage d'ordre des contraintes,
- PHB Scheduling class (PSC): un ensemble des PHB appliqués aux différentes Behaviour Aggregates du même Ordered Agregate.

Par exemple, AF_{1x} est un PSC comprenant les PHB de AF_{11}, AF_{12} et AF_{13}. De même, EF est un PSC comprenant exclusivement le PHB de EF.

En effet, MPLS permet à l'administrateur réseau de spécifier comment les agrégats DiffServ (DiffServ Behavior Aggregates: BA) sont affectés aux LSPs. La question est de savoir comment affecter un ensemble de BA au même LSP ou à des LSP différents. Quand les paquets arrivent à un Ingress du domaine MPLS, le champ DSCP est ajouté à l'entête qui correspond au Behavior Aggregate (BA). A chaque noeud traversé, le DSCP (Differentiated Services Code Point) permet de sélectionner le PHB (Per Hop Behavior), qui définit le Scheduling (classe prioritaire, WFQ, CBQ,...) et les probabilités de rejet du paquet. Cette correspondance entre les classes DiffServ et le PHB de MPLS offre de multiples possibilités à un administrateur pour un coeur de réseau DiffServ/MPLS. L'affectation de classes DiffServ à différents LSPs permet aussi d'attribuer des mécanismes de protection différents pour ces différentes classes de service. On distingue deux grands types de LSP que sont les E-LSP (EXP-Inferred-PSC LSP) et les L-LSP (Label-Only-Inferred-PSC LSP). Les E-LSP et L-LSP, établis soit par LDP [1] soit par RSVP [1], peuvent coexister dans le même réseau DiffServ/MPLS. Pour plus d'informations sur l'établissement des chemins avec le DS-TE, se référer à [72].

1.3 Modèles devéloppés au LAGIS pour MPLS-TE

Dans ce paragraphe, on étudie deux modèles (LBWDP[59] et PEMS[58]) [56] de routage dynamique par équilibrage de charge sur plusieurs chemins traversant un réseau de fournisseur d'accès internet (FAI). Ces deux modèles ont été développés au sein du laboratoire LAGIS [55], dans le cadre de la thèse soutenue par K. LEE [56]. Dans [60][57][56], des comparaisons ont été faites avec d'autres modèles d'ingénierie de trafic comme MATE [28] ou LDM [87], et les résultats ont montré que LBWDP et PEMS semblent plus performants.

1.3.1 LBWDP: Load Balancing over Widest Disjoint Paths

LBWDP [59] est un modèle hybride pour l'ingénierie de trafic par équilibrage de charge basé sur le principe du routage multi-chemins. Comme nous l'avons vu précédemment dans le routage multi-chemins, LBWDP comprend deux étapes : une étape de sélection des chemins candidats effectuée par l'algorithme WDP (Widest Disjoint Path) [74] et une étape de répartition du trafic faite par l'algorithme PER (Prediction of Effective Repartition) [60][56].

La sélection des chemins de WDP (Widest Disjoint Path) [74] est essentiellement basée sur 2 métriques: la largeur et la longueur d'un chemin. La largeur d'un chemin est un moyen pour détecter les goulots d'étranglements dans le réseau et les éviter si possible. Par contre, la longueur du chemin, contrairement à la majeure partie des autres approches utilisées dans l'internet, n'est pas considérée comme une mesure directe en fonction du nombre de sauts. WDP exprime la métrique de la longueur d'un chemin en fonction

du taux d'utilisation des liens que comporte le chemin. WDP réalise l'étape de sélection des chemins candidats en se basant sur le calcul de la largeur de bande passante résiduelle de l'ensemble des chemins candidats. On sélectionne ainsi un ensemble de chemins disjoints, les plus courts du point de vue nombre de sauts et dont la largeur résiduelle est supérieure à une limite fixée par l'administrateur. Un chemin est ajouté à l'ensemble des chemins candidats si cela permet d'accroître la largeur de cet ensemble. Par contre, un chemin est supprimé de cet ensemble si cela ne réduit pas sa largeur.

Pour l'étape de la répartition du trafic, LBWDP utilise l'algorithme PER (Prediction of Effective Repartition) [60][56]. PER constitue une amélioration de la partie de répartition de trafic de l'algorithme LDM [87]. Le choix d'un chemin lors de l'étape de répartition de trafic avec LDM est basé sur une probabilité de distribution en fonction des critères comme la longueur en nombre de sauts et le taux d'utilisation des chemins. L'inconvénient de cette distribution est qu'elle utilise les mêmes informations sur le trafic pour toute une période donnée. En effet, en appliquant une comparaison entre deux taux de répartition effectif et objectif, les deux basés sur des critères de performances du réseau, PER offre plus de maniabilité et de dynamique lors de l'étape de répartition du trafic. Il utilise l'information sur la différence entre la répartition théorique de trafic et celle du taux d'utilisation effective des chemins, afin de choisir le chemin à emprunter par le trafic. Il comprend lui-même deux sous-étapes.

Lors de la première étape, il calcule un paramètre de sélection S_i pour la répartition du trafic. Ce paramètre est calculé par la différence entre le trafic désiré ou objectif r_i et le trafic effectif e_i selon la relation (1.1).

$$S_i = \frac{r_i - e_i}{r_i} \qquad Avec\ e_i \neq 0 \tag{1.1}$$

Cependant, le taux de répartition objectif r_i est fonction du nombre de sauts et de la capacité résiduelle du chemin. Il est calculé par la relation (1.2).

$$r_i = p_0 \frac{H}{hc(i)} + p_1 \frac{b(i)}{B} \qquad Avec\ p_0 + p_1 = 1 \qquad (1.2)$$

$$H = \frac{1}{\sum_{i=1}^{k} \frac{1}{hc(l_i)}} \qquad et \qquad B = \sum_{i=1}^{k} b(i)$$

Avec *hc(i)*: la longueur d'un chemin en nombre de sauts, et *b(i)* est la bande passante residuelle du chemin, pour *i=1,..., k* l'ensemble des chemins candidats.

Tandis que le taux de répartition effectif e_i est exprimé par le rapport entre la quantité de trafic réellement distribuée et la somme des demandes depuis la dernière mise à jour (link state update) (relation 1.3).

$$e_i = \frac{Trafic\ effectif\ sur\ le\ chemin}{Demande\ totale\ durant\ la\ periode} \qquad (1.3)$$

Dans la deuxième étape, PER sélectionne le chemin qui a la plus grande valeur positive pour le paramètre de sélection S_i, et possédant suffisamment de bande passante résiduelle pour acheminer la totalité de la demande courante. Le fait de choisir une valeur positive de S_i signifie que le taux de répartition réel e_i est inférieur à l'objectif calculé r_i. Donc, plus S_i est grand, plus le choix du chemin est meilleur. Pour plus d'informations sur l'algorithme PER, se référer aux [60][56].

1.3.2 PEMS: PEriodic Multi-Step routing algorithm for DS-TE

L'IETF [48] s'est intéressé ces dernières années au déploiement simultané de MPLS-TE (Trafic Engineering based on MPLS) et de DiffServ (Differen-

tiated Services) afin de tirer avantage de ces deux techniques d'amélioration de la qualité de service dans les réseaux filaires. L'intégration de ces deux technologies corresponds à la catégorie des modèles DS-TE (Diffserv aware Traffic Engineering). Différentes approches de DS-TE sont possibles. Dans ce cadre, un nouveau algorithme appelé PEMS a été proposé dans [58] [56]. Il est utilisé pour favoriser les différents besoins des utilisateurs et pour fournir des services différenciés pour les trois classes de trafic d'un réseau DS-TE. Son principe général consiste à mettre en oeuvre l'équilibrage de charge par classes de trafic tel que défini par Diffserv.

L'objectif de PEMS [58] est donc de développer une méthode de routage qui effectue de l'équilibrage de charge par classe de trafic au niveau de chaque routeur d'entrée dans un réseau FAI. Les trois classes traitées par PEMS sont : le Service Premium (EF: Expedited Forwarding) pour le trafic sensible au délai comme le trafic VoIP (Voix sur IP), le Service Garanti (AF : Assured Forwarding) pour le trafic qui nécessite une garantie de bande passante (la vidéo) ou des taux d'erreurs garantis, et le service correspondant au trafic au mieux (BE : Best-Effort) pour des applications comme le transfert de fichiers ou la messagerie électronique. Le but principal de PEMS est donc de permettre à chaque trafic d'être acheminé selon ses exigences de qualité de service (QdS).

Le modèle PEMS comporte trois étapes (figure 1.18) : l'étape du prétraitement, celle de la sélection des chemins candidats et celle de la répartition du trafic sur les LSP (Label Switched Path).

L'étape de prétraitement s'effectue hors ligne et consiste à choisir les chemins les plus courts en nombre de sauts entre une source et une destination. Ensuite elle en extrait les maximums disjoints possible. L'objectif est d'évi-

FIG. 1.18 – *Les 3 étapes de PEMS*

ter une recherche combinatoire en ligne. Elle est basée uniquement sur la topologie.

La deuxième étape s'effectue en ligne et consiste à sélectionner des chemins candidats selon les trois classes de trafic EF, AF et BE. Elle tient compte des critères de bande passante résiduelle et de délai mesuré pour sélectionner des sous-ensembles réduits de chemins fournis par l'étape de prétraitement. En effet, les différents chemins de l'étape de prétraitement sont triés en fonction du délai mesuré et de la bande passante résiduelle pour les classes EF et AF respectivement. Ensuite, un sous ensemble de chemins est choisi pour assurer le transfert de chaque classe de trafic correspondante. Elle choisit un certain nombre de chemins qui ont les meilleurs délais pour la classe EF, un certain nombre de chemins qui ont les meilleures bandes passantes résiduelles pour la classe AF, et le reste sera affecté à la classe BE. A chaque nouvelle mise à jour des paramètres de qualité de service (Link State Update effectué par les routeurs voisins), l'ensemble des chemins candidats est recalculé afin de considérer toujours un ensemble limité de meilleurs chemins. Le nombre de chemins candidats retenus et la périodicité des mises à jour sont des paramètres fixés par l'administrateur réseau.

Pour la dernière étape, PEMS utilise l'algorithme PER [60][56] en l'adaptant aux différentes classes de trafic. Un nombre de chemins candidats est

associé à chaque classe de trafic. Ce nombre est fixé par l'administrateur réseau. Par exemple, sur 10 chemins, ont peut affecter les 4 ayant les meilleurs délais pour acheminer le trafic de la classe prémium. Les suivants sont affectés à la classe AF et les derniers sont associés au trafic BE. Pour chaque classe de trafic, PER calcule le taux théorique de répartition sur chaque chemin de la classe. Ce taux est transformé en un taux de répartition relatif qui tient compte de la répartition effective des demandes sur les chemins considérés. Lorsqu'une demande d'une classe donnée arrive, elle est affectée au chemin ayant le taux de répartition relatif le plus grand. En effet, plus ce taux est élevé plus le chemin est sous-utilisé par rapport au plan de routage établi pour la période donnée. L'équilibrage de charge effectué par PER consiste à affecter toute nouvelle demande dans sa classe, au chemin le moins utilisé à l'instant donné sous réserve que son taux résiduel d'utilisation de bande passante soit compatible avec le débit requis par la demande. Si ce n'est pas le cas, le routeur force une mise à jour des paramètres de qualité de service (délai et bande passante résiduelle) avant la fin de la période en cours. Dans ce cas, on relance l'étape de calcul des chemins candidats et on recalcule les taux.

L'objectif de PEMS [58] est de minimiser l'utilisation maximale des liens exactement comme LBWDP [59] tout en différentiant les chemins en fonction de la qualité de service requise par la classe de trafic de la demande courante.

La figure 1.19 présente certaines caractéristiques de LBWDP et PEMS en termes de structure et d'objectif. Elle apparaît comme une comparaison entre les fonctionnalités de ces deux modèles.

	LBWDP		PEMS	
Equilibrage de charge	Routage multi-chemins		Routage multi-chemins	
Service différencié	Non		3 classes : EF, AF, BE	
Nombre d'étapes	Hors ligne	En ligne	Hors ligne	En ligne
	0	2	1	2
Combinaison (Hybride)	WDP + PER		Max chemins disjoints + Chemins candidats + PER	

FIG. 1.19 – *Caractéristiques de LBWDP et PEMS*

1.4 Conclusion

Dans ce chapitre, nous avons présenté le concept général de l'ingénierie de trafic dans les domaines des réseaux IP et IP/MPLS. Nous avons détaillé le mécanisme de MPLS-TE introduit dans les réseaux IP, reposant sur le principe du routage par contrainte, en se basant sur l'approche d'équilibrage de charge par routage multi-chemins, tout en intégrant des contraintes de qualité de service QdS. En effet, MPLS-TE permet, par un contrôle des chemins empruntés, d'optimiser l'utilisation des ressources et de réduire les risques de congestion dans les réseaux. Les technologies MPLS-TE sont basées essentiellement sur trois fonctions de base du routage par contrainte, incluant la découverte de la topologie (TE), le calcul des chemins explicites, ainsi que l'établissement des LSPs. Dans ce cadre, nous avons présenté deux modèles pour l'ingénierie de trafic dans un réseau IP/MPLS, à savoir LBWDP et PEMS. Ceci nous amène au chapitre suivant, où nous évaluons ces deux modèles sur des topologies similaires à Internet, pour mesurer leurs performances et tester leur capacité à être déployé sur des réseaux de grande taille.

Chapitre 2

Etude comparée de modèles de routage pour l'ingénierie de trafic

L'objectif de ce chapitre est l'élaboration d'un ensemble de métriques pour l'évaluation de performances de deux modèles de routage multi-chemins LBWDP [59] et PEMS [58] présentés dans le chapitre précédent. Afin de mener à bien cette tâche, nous abordons le problème au niveau organisationnel, d'implantation et fonctionnel. Les deux modèles proposés permettent de traiter la problématique du routage adaptatif dans un réseau à commutation de paquets de type IP/MPLS. Ce type de réseau est caractérisé par des conditions de trafic dynamiques, ce qui nécessite de prendre en compte la QdS au niveau du routage. Par conséquent, toute approche du routage adaptatif doit être suffisamment réactive et robuste pour prendre en compte toute modification des conditions de trafic. Cependant, l'efficacité de ces approches dépend fortement des informations sur la charge du réseau. Ces informations doivent être suffisantes et pertinentes et en même temps refléter de manière fiable la charge réelle du réseau au moment de la prise de décision de routage. Ce chapitre traite la mesure des performances de ces deux modèles. L'objectif

est d'évaluer les différents paramètres caractérisant leur plan de routage et la qualité de service offerte après leur mise en oeuvre sur des grandes topologies. Et ainsi mesurer l'impact de leur complexité respective et donc la capacité à les déployer sur des réseaux de grande taille (scalabilité).

2.1 Introduction aux méthodes de simulation et d'évaluation de performances

La Qualité de Service (QdS) est caractérisé essentiellement par la bande passante, le délai de bout en bout, la gigue et le taux de perte des données. Au fur et à mesure de l'évolution de l'Internet et de ses applications, les utilisateurs sont devenus de plus en plus exigeants et demandent à leurs fournisseurs d'accès des garanties de service (Service Level Agreement). Il est devenu alors nécessaire de connaître l'état courant des performances du réseau et de la qualité de service fournie, afin d'assurer la bonne supervision du contrat liant les deux parties. Ces nouveaux besoins ont considérablement contribué à l'émergence d'un nouveau domaine de recherche qui est l'évaluation de performances dans les réseaux. L'évaluation de performances [82] est une étape importante dans la compréhension d'un système d'information lors de sa conception, son implémentation, son déploiement, son utilisation et son évolution.

Le terme *performance* désigne la mesure chiffrée du fonctionnement d'un système ou de l'un de ses composants lors de la réalisation d'une tâche qui lui a été confiée. Les différents types d'évaluation de performances partagent certaines étapes communes, comme:

- *Identifier l'objectif de l'étude et définir les limites du système:* dans un même environnement matériel et logiciel, la définition du système dépend

de l'objectif de l'évaluation. La définition des limites du système affecte les métriques de performance utilisées pour comparer les systèmes.

- *Sélectionner les métriques de performance:* les critères pour comparer la performance sont appelés métriques. Généralement, ces métriques sont relatives à la rapidité, la précision et la disponibilité des services. Par exemple, la performance d'un réseau est souvent mesurée en terme de débit, de délai (rapidité), et de taux d'erreur (précision).

- *Sélectionner la technique d'évaluation:* il existe plusieurs approches d'évaluation de performances des systèmes. Parmi ces approches, on trouve les modèles analytiques, les simulations, les mesures.. Le choix d'une approche ou d'une autre dépend de plusieurs critères [82]. Ces approches peuvent être classées en deux grandes catégories (Fig. 2.1): les approches basées sur la mesure et celles basées sur la modélisation.

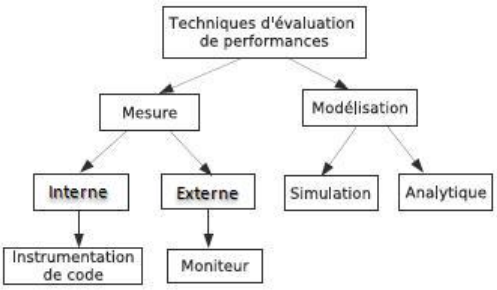


FIG. 2.1 – *Classification des techniques d'évaluation de performances*

La première catégorie permet de quantifier les critères de performance en les mesurant directement sur un système réel. Dans ce cas, nous distinguons deux variantes d'approches des mesures: l'instrumentation du code source de l'application en insérant des codes de mesure (in-side), ou l'utilisation de moniteurs externes à l'application pour quantifier les mesures au cours de cycle d'exécution de l'application (out-side).

La deuxième catégorie est basée sur la modélisation, elle permet de spécifier les caractéristiques du système à évaluer afin de prédire ses performances. Les modèles élaborés par cette catégorie sont utilisés pour effectuer des simulations ou des études analytiques du système. Chacune de ces deux méthodes suit un fonctionnement différent. Les simulations mettent en oeuvre des modèles conceptuels du système, qui nécessitent leurs développements dans un outil de simulation. En revanche, la deuxième méthode (l'étude analytique) met en oeuvre des modèles approximatifs (files d'attente, réseaux de Petri,..) sous la forme de formules mathématiques ou tout autre formalisme.

Pour réaliser notre étude, nous avons opté pour l'évaluation par simulation.

La simulation est une technique souvent utilisée pour évaluer les performances des systèmes informatiques. Elle constitue un moyen utile pour prédire les performances d'un système et les comparer sous plusieurs de ses configurations. Un atout majeur de cette technique est sa flexibilité, puisqu'elle permet d'évaluer le système sous plusieurs conditions et configurations. Son inconvénient est qu'elle nécessite un temps potentiellement important et des ressources de calcul conséquentes. Les étapes générales de l'évaluation par simulation sont décrites ci-dessus. Elles sont illustrées en détails dans la Fig.2.2.

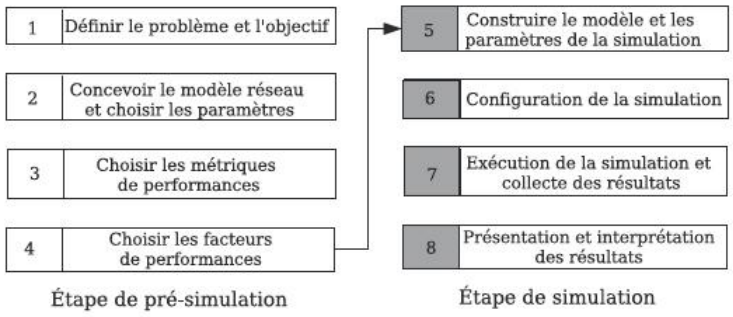


FIG. 2.2 – *Les étapes de réalisation d'une simulation*

2.2 Objectifs de la simulation

Les deux modèles LBWDP et PEMS ont été proposés pour résoudre le problème des déséquilibres de charge engendré par le problème de routage du plus court chemin utilisé dans les réseaux IP, tout en offrant une garantie de qualité de service QdS. Leur principal objectif est de contrôler la qualité de service de bout-en-bout, en imposant une utilisation efficace des ressources du réseau. Une comparaison de ces deux modèles avec d'autres modèles de l'ingénierie de trafic a été réalisée dans [56]. Elle a montré que ces deux modèles semblent plus performants que d'autres comme MATE [28], WDP avec PER [74] ou LDM [87]. En effet, cette comparaison a été effectuée sur des topologies simples de tailles et de complexité qui ne reflètent pas la réalité. Nous proposons d'étendre l'étude à des topologies plus complexes (plusieurs centaine de routeurs) et suivant différents scénarios qui s'approchent du réseau réel. Le but est de mesurer l'impact de leur complexité respective et donc la capacité à les déployer sur des réseaux de grande taille (scalabilité). Les principaux objectifs de ce chapitre sont donc d'étudier les différents caractéristiques de ces deux modèles et d'effectuer une analyse comparative de leurs performances, et on pourra ainsi évaluer leurs degrés d'efficacité pour la mesure de l'ingénierie de trafic et de la qualité de servise QdS (équilibrage de charge, délai, taux de perte,..). Enfin, on mesurera leur capacité de passage à l'échelle en les testant sur de grandes topologies similaires à celles d'internet, afin de mesurer l'impact de la densité du réseau sur leurs performances.

2.3 Critères d'évaluation

La mesure de la qualité de service dans l'Internet trouve son origine dans les travaux réalisés par l'IETF [48] au début de l'année 1996. Un groupe de

l'IETF composé de chercheurs et de fournisseurs de services a été chargé de définir des recommandations sur les mesures de performances pour différentes technologies. Le groupe avait pour objectifs de définir la terminologie associée aux critères de base, de définir des méthodologies de mesure en vue d'offrir des évaluations de performances standards pouvant être utilisées par les différents FAIs, et finalement de développer un ensemble standard de critères, commun aux utilisateurs et aux FAIs, caractérisant la qualité, la performance et la fiabilité d'un service IP. Le but des critères d'évaluation est donc d'établir une base commune de connaissance au niveau des performances et de la fiabilité du réseau afin d'en obtenir une connaissance précise. Dans ce chapitre, on s'interesse à l'évaluation des critères de routage offerts par les algorithmes ainsi à leur scalabilité à être déployés sur des réseaux de compléxité similaire à celle d'internet.

2.3.1 Qualité de routage

Un critère est une quantité soigneusement définie exprimant un niveau de performance directement lié au fonctionnement et à la fiabilité de ce réseau. En effet, un critère de qualité de service QdS doit être utile à la fois pour les utilisateurs et pour les fournisseurs d'accès en leur permettant de bien quantifier les performances du réseau qu'ils offrent (les fournisseurs) ou qu'ils demandent (les utilisateurs). Les critères de base définies par l'IETF sont:

- La mesure de la connectivité entre deux équipements,
- Le délai unidirectionnel (One-Way Delay),
- Le taux de pertes unidirectionnelles de paquets (One-Way Loss),
- Le délai et le taux de pertes de paquets aller-retour (Round-Trip Delay and Loss).

D'autres critères plus complexes sont élaborés tels que:
- La variation de délai (Gigue),
- Les modèles de perte de paquets (Loss Patterns),
- Le réordonnancement des paquets (Packet Reordering).

Par contre, L'utilisation d'une métrique unique peut s'avérer insuffisante pour satisfaire les besoins de certaines applications. Par exemple, une application de visioconférence peut requérir d'une part, une certaine bande passante afin que la qualité de la retransmission vidéo soit correcte et d'autre part, un délai de bout en bout borné pour que les conversations ne soient pas entrecoupées de "blancs"' désagréables pour les utilisateurs. Il est donc possible de considérer plusieurs critères séparément. Cependant, il est prouvé dans [90] que le problème consistant à trouver un chemin satisfaisant à deux contraintes parmi les suivantes : délai, taux de perte, coût et gigue, est NP-complet. Les combinaisons restantes sont donc celles associant la bande passante au délai, au taux de perte ou à la gigue. C'est dans ce cadre, qu'on a choisi les paramètres de qualité de service essentiels, à savoir l'utilisation moyenne et maximale des liens du réseau, et le délai moyen de bout en bout.

L'utilisation des liens

En théorie, la bande passante désigne la différence en Hertz entre la plus haute et la plus basse des fréquences utilisables sur un support de transmission. Dans la pratique, ce terme désigne le débit d'une ligne de transmission, calculé en quantité de données susceptibles de transiter dans une unité de temps donnée. En revanche, l'utilisation d'un lien est définit comme étant la bande passante utilisée, donc une mesure de l'utilisation de la bande passante du lien. L'utilisation moyenne permet d'indiquer le niveau d'équilibrage de charge dans le réseau. Plus il est faible plus le modèle effectue un bon équi-

librage du trafic. Par contre, l'utilisation maximale des liens nous renseigne sur le risque de congestion dans le réseau.

Le délai de bout en bout

Il s'agit de la durée nécessaire à l'acheminement d'un paquet de données de bout en bout. Cette durée dépend de la qualité du support des liaisons, mais aussi du temps passé dans les différentes files d'attente du chemin. Par conséquent, le délai augmente avec la charge du réseau. Le délai est composé du délai de transmission, de propagation et d'attente dans les buffers. Le délai de propagation est estimé selon le rapport entre la distance et la vitesse de propagation. Le délai de transmission est calculé selon le rapport entre la taille du paquet et la bande passante. Le délai d'attente est égal au temps de séjour moyen de chaque paquet dans la file d'attente du routeur. Il sera calculé lors de l'étape de simulation en utilisant le service "queue monitoring " (Fig. 2.3) du logiciel de simulation réseaux NS-2. Le délai de propagation et le délai de transmission sont des délais statiques. Par contre, le délai d'attente est fonction de la charge. Donc l'allongement du délai de transfert moyen de bout en bout permet de mesurer indirectement la charge du réseau. En revanche, le délai de bout en bout permet de fournir une indication sur la capacité de l'algorithme à différencier le délai selon les classes de trafic. La corrélation des délais moyens obtenus en fonction de l'utilisation des liens est une analyse permettant de savoir si l'équilibrage de charge permet d'optimiser le délai moyen.

2.3.2 Scalabilité

La scalabilité est la capacité du système à s'adapter à un nouvel environnement sans mettre en cause ses performances. Elle représente sa capacité à

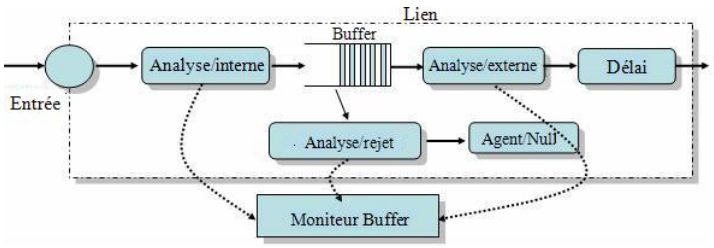


FIG. 2.3 – *Le support "queue monitoring" dans le simulateur NS*

fonctionner correctement et à évoluer en cas de montée en charge. Elle est nécessaire lorsqu'on veut s'adapter aux caractéristiques variables des ressources manipulées, ainsi qu'à la diversité des services visés. En effet, un système est dit scalable si l'augmentation de processus entraîne une augmentation proportionnelle des performances, car il pourrait offrir un redimensionnement selon les besoins des applications. La scalabilité permet un meilleur passage à l'échelle en minimisant la complexité calculatoire et en offrant un faible coût et accès mémoire de façon à permettre une utilisation "temps réel". Dans ce chapitre, on s'intéresse à la scalabilité des deux algorithmes LBWDP et PEMS au niveau réseau c.à.d. à leur capacité à être déployés sur des réseaux de grandes tailles avec une compléxité similaire à internet. A cette fin, nous nous intéressons à l'utilisation de générateurs de topologies pour l'obtention automatique de modèles de grande taille.

2.4 Génération de topologies

Cartographier l'Internet est un enjeu majeur pour la recherche. Par exemple, une connaissance précise de la topologie de l'Internet permettrait de créer un outil de modélisation capable de générer des graphes réalistes utilisables dans les simulations des protocoles réseaux. Internet, contrairement aux réseaux qui le composent, n'a pas d'autorité définissant l'évolution de sa topologie.

C'est pourquoi personne n'est capable de fournir aujourd'hui une carte détaillée de l'Internet. Cela constitue un défi à relever car beaucoup de créateurs de protocoles apprécieraient une telle information.

2.4.1 Modèles pour la génération de topologies

Générer une topologie réseau est le premier pas à franchir dans la création d'un scénario de simulation d'un protocole réseau. Les résultats de simulation dépendent souvent de la topologie utilisée, particulièrement dans le cas des protocoles de routage et des protocoles multipoints. Par conséquent les topologies réseaux doivent être générées avec la plus grande précision possible. De plus, les protocoles destinés à être déployés dans l'Internet devraient être testés sur des topologies réseaux ressemblant à celle d'Internet. En effet, les outils de simulation nécessitent des topologies de réseaux en entrée. Le réalisme de ces topologies est important pour que les résultats des simulations soient significatifs. Les distances entre les noeuds par exemple ont un impact sur les délais de transmission. Le nombre de plus courts chemins entre les noeuds influe sur la fiabilité des communications. Enfin, le degré des noeuds joue un rôle crucial dans la diffusion multipoint et influe sur les phénomènes d'inondation du réseau.

Une topologie de réseau est habituellement modélisée par un graphe non orienté où les sommets sont des entités de communication, les arêtes des liens de communication. Un logiciel qui crée des topologies de réseaux est habituellement appelé un générateur de topologies. Les propriétés des graphes telles que le degré moyen et le diamètre sont appelées propriétés topologiques.

Nous nous intéressons ici à des graphes composés de routeurs et de liens uni-directionnels pour les simulations d'algorithmes LBWDP et PEMS, destinés à un environnement IP/MPLS.

Les générateurs de topologies existants utilisent diverses méthodes de création de topologies, on note par exemple:
- Le placement aléatoire qui consiste à placer les sommets de façon aléatoire et à les connecter en utilisant une fonction probabiliste liée à la distance séparant les sommets,
- La croissance incrémentale et la connectivité préférentielle qui consistent à ajouter les sommets au fur et à mesure en les connectant de préférence à des sommets ayant un degré élevé,
- L'ingénierie inverse (reverse engineering) qui consiste à construire une distribution appropriée des degrés des sommets du graphe puis à attribuer ces degrés aux sommets.

L'un des plus anciens modèles topologiques fut créé par Waxman [91]. Il est appelé modèle topologique plat. Les noeuds sont placés aléatoirement dans un plan euclidien sans tenir compte d'aucune hiérarchie entre eux. Il prend en données le nombre de noeuds N, les paramètres α et β (par défaut, ils sont placés à 0.25 et 0.20 respectivement), et il applique une fonction probabiliste pour décider de positionner une liaison entre deux noeuds par une arête. Ensuite, en ajustant les deux paramètres α et β, les caractéristiques des graphes sont modifiés et il est ainsi possible d'obtenir des graphes modélisant des interconnexions réelles. La probabilité P pour avoir une arête entre deux noeuds u et v est exprimée par la relation (2.1):

$$P(u,v) = \alpha\, e^{-\frac{d(u,v)}{\beta L}} \tag{2.1}$$

où $d(u,v)$ est la distance euclidienne entre u et v dans le plan, L est la distance maximale entre 2 noeuds, α est le degré de connexion moyen d'un noeud et β est la longueur moyenne d'une arête. Ensuite, une valeur aléatoire est générée entre 0 et 1. Une arête est crée entre les deux noeuds u et v si la valeur ainsi générée est plus petite que $P(u,v)$. Pour fixer les paramètres α et

β, il faut tenir compte du fait que plus α augmente plus le nombre d'arêtes augmente, et plus β augmente plus le rapport entre arêtes longues et arêtes courtes augmente. Le modèle de Waxman a été modifié (Fig. 2.4) en ajoutant le "scale degree" à la probabilité P lié au nombre de noeuds, pour éviter que le nombre d'arcs n'augmente quand le nombre de noeuds augmente. Cette technique consiste à balancer l'exponentiel de P par le nombre d'arcs associé à un noeud.

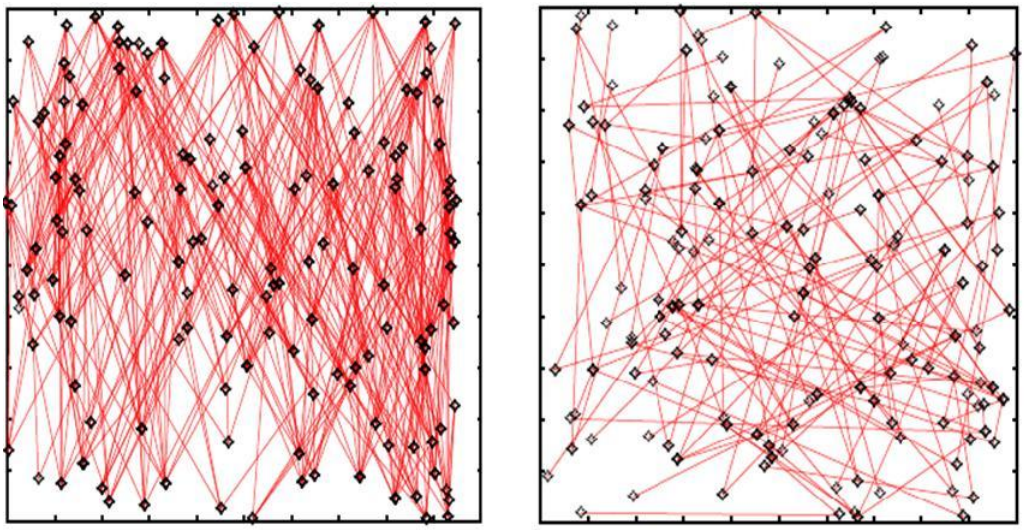


FIG. 2.4 – *Génération d'une topologie Waxman avec les 2 techniques*

Le modèle Waxman fut plus tard remplacé par des modèles hiérarchiques tels que Tiers [26] et GT-ITM (Transit-Stub)[16]. Ces modèles essayent de recréer la hiérarchie à plusieurs niveaux que l'on peut trouver dans Internet. En effet, Internet peut être structuré en différents niveaux hiérarchiques en prenant en compte ses noeuds (hôtes et routeurs), ses réseaux (réseau d'une organisation) et ses systèmes autonomes.

La découverte des lois de puissance dans l'Internet par Faloustos et al. [29] a entraîné la création d'un nouveau type de modèle topologique. Ce

dernier, que nous avons logiquement appelé modèle topologique des lois de puissance, comme Inet [53](générateur de topologies AS), tente de modéliser toutes, ou une partie, des lois de puissance découvertes dans Internet. Ainsi la distribution des routeurs en fonction de leur degré (i.e., le nombre de liens incidents) obéit à une loi de puissance. La moitié des routeurs ont un degré de 1, le quart à un degré de 2 et ainsi de suite. A l'autre bout de la distribution, quelques routeurs ont des degrés très élevés. Cette propriété diffère des graphes aléatoires dans lesquels la distribution des noeuds en fonction de leur degré obéit à une loi de Poisson.

2.4.2 BRITE, un exemple de générateur de topologies

BRITE [13] semble être actuellement un des meilleurs outils fonctionnels. Il est conçu pour être flexible. Comme l'illustre la figure 2.5, BRITE supporte la génération de différentes catégories de modèles. En effet, il peut générer une topologie d'interconnexion de systèmes autonomes (AS) ou d'interconnexion de routeurs. Il peut également intégrer ces deux topologies dans une modélisation hiérarchisée (le système autonome est constitué de routeurs interconnectés). Lors de la génération, on peut choisir l'une des méthodes de placement des noeuds suivante:

- Aléatoire: placement des noeuds d'une façon aléatoire dans le plan
- Heavy-tailed : diviser le plan en carrés, et placer un nombre de noeuds choisi selon une distribution Heavy-tailed aléatoirement dans chaque carré. Ainsi on peut définir les propriétés de la méthode d'interconnexion utilisée (Waxman ou Barabt'asi-Albert).

NB: La méthode Barabt'asi-Albert consite à connecter les noeuds selon une approche incrémentale. Quand un noeud i joint le réseau, la probabilité qu'il soit connecté à un noeud j déjà existant dans le réseau est calculé par la

relation (2.2):

$$P(i,j) = \frac{d_j}{\sum_{k \in V} d_k} \tag{2.2}$$

avec d_j est le degré du noeud ciblé, et V l'ensemble de noeuds ayant déjà joints le réseau.

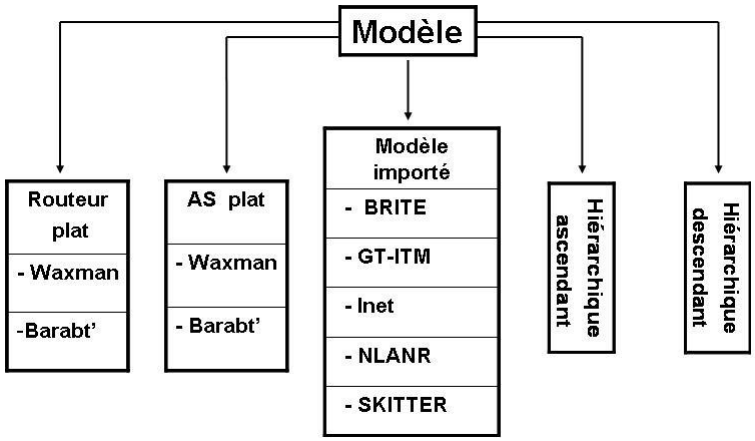


FIG. 2.5 – *Structure générale de BRITE*

La figure 2.6 présente l'architecture principale de BRITE. L'application lit les paramètres de génération depuis le fichier de configuration (1) qui peut être généré manuellement par l'utilisateur ou automatiquement à l'aide de l'interface graphique (GUI) de BRITE (Fig. 2.7). BRITE possède également la capacité d'importer des topologies générées par d'autres générateurs (GT-ITM [16], Inet [53], Tiers [26]), ou des topologies extraites directement de l'internet (NLANR [76], Skitter [21]). En sortie, BRITE possède son propre format propriétaire et peut également fournir la topologie du réseau au format du simulateur de réseau NS-2 [77]. BRITE intègre également un outil d'analyse appelé BRIANA. Cet outil fournit un ensemble d'analyses monotones qui peuvent être appliquées à une topologie importée dans BRITE.

La génération d'un modèle avec BRITE comprend quatre étapes:

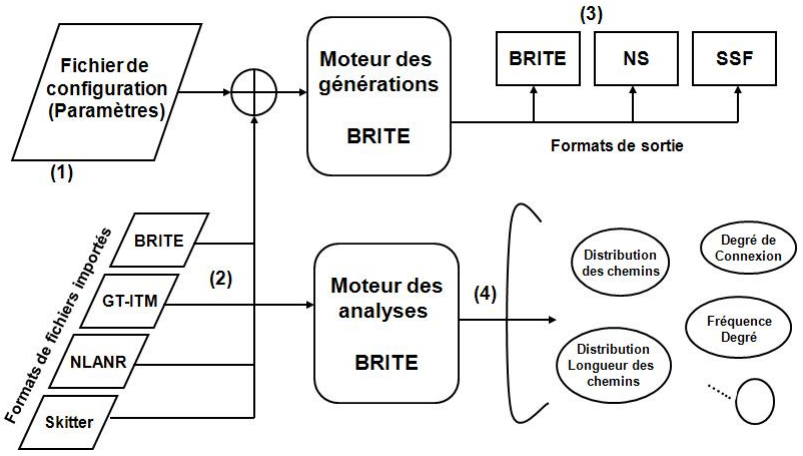


FIG. 2.6 – *Architecture générale de BRITE*

1. Placement des noeuds dans le plan,

2. Connexion des noeuds les uns aux autres,

3. Affectation des valeurs aux attributs des composants de la topologie. Ces attributs sont notamment, le délai et la bande passante de chaque lien,

4. Génération de la topologie selon le format spécifié.

BRITE est implémenté avec deux langages de programmation (Java et C++) et peut être étendu pour incorporer de nouveaux modèles de topologies.

2.4.3 Comparaison des générateurs de topologies

Dans cette section nous proposons une étude comparative de quelques générateurs de topologies de type Internet (BRITE, Inet, GT-ITM, Tiers). Nous essayons d'évaluer leurs qualités et leurs défauts et nous proposons une grille de notation pour tous ces générateurs. Notre objectif est de choisir le meilleur générateur permettant de modéliser le plus fidèlement possible la topologie d'internet. Pour cela, nous avons évalué et comparé la précision

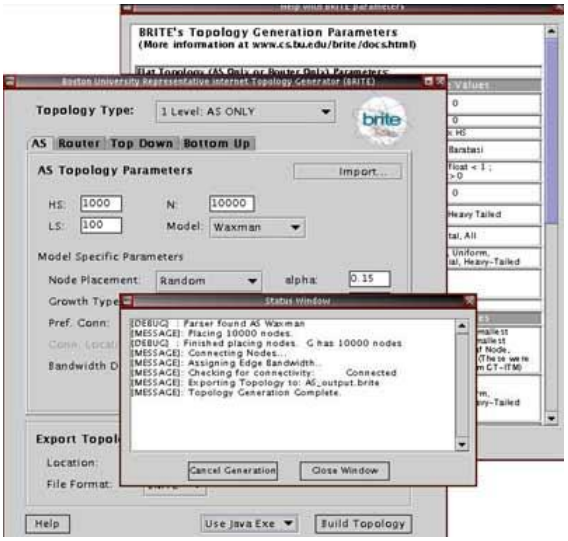


FIG. 2.7 – *Format de l'interface graphique de BRITE*

de ces générateurs en mesurant certaines propriétés topologiques des graphes qu'ils génèrent, comme le niveau de la topologie générée (Autonomous System (AS) ou niveau des routeurs), et la nature de la topologie (Hiérarchique ou basée sur le degré). Les deux tableaux donnés par la figure 2.8 montrent que BRITE est le meilleur générateur en termes de corrélation avec une topologie réelle aussi bien au niveau représentation (niveau AS ou routeur) (Fig 2.8.a) qu'au niveau de la structure (hiérarchique ou basée sur le degré) (Fig. 2.8.b), car il est le seul capable de générer des topologies selon ces différentes caractéristiques. BRITE doit ce résultat au fait qu'il implémente la totalité des fonctions de génération de topologies contrairement aux autres générateurs.

2.5 Modèle de Simulation

La validation et l'évaluation des modèles de routage multi-chemins proposés nécessitent leurs implémentations sur un simulateur. Une partie rela-

Générateurs	Niveau AS	Niveau Routeur
BRITE	Oui	Oui
Inet	Oui	Non
GT-ITM	Non	Oui
Tiers	Non	Oui

(a)

Générateurs	Hiérarchique	Basé sur le degré
BRITE	Oui	Oui
Inet	Non	Oui
GT-ITM	Oui	Non
Tiers	Oui	Non

(b)

FIG. 2.8 – *Comparaison des générateurs de topologies*

tivement importante de ces travaux de thèse a été consacrée à cette tâche, notamment à la prise en main de l'outil de simulation NS-2 [77] et à la méthodologie de création, d'implémentation et de validation de modèles de routage. Dans ce paragraphe, nous passerons en revue quelques outils de simulation couramment utilisés. Nous nous focaliserons ensuite plus particulièrement sur l'outil NS-2 en développant l'implémentation de deux algorithmes LBWDP et PEMS sur lesquels s'appuie notre étude.

2.5.1 Outils de simulation

Il existe de nombreux outils de simulation permettant l'implémentation et l'évaluation des performances de protocoles. Certains d'entre eux sont des prototypes issus de la recherche universitaire et d'autres des produits commerciaux. Les deux principaux outils sont le Network Simulator version 2 ou NS-2 [77] qui est un logiciel libre, et OPNET [79] qui est un produit commercial. Les différences majeurs entre ces deux outils sont essentiellement l'interface graphique conviviale de OPNET qui manque dans NS-2 et la documentation conséquente de OPNET. Ces deux outils représentent les outils standards pour toute étude basée sur la simulation dans le domaine du réseau. Des études comparatives ont été faites dans [62] portant sur le degré

de précision de ces deux outils dans le cadre de la simulation du réseau au niveau paquet, et elles ont montré que d'un point de vue recherche, les résultats de NS-2 et OPNET sont similaires. Par contre, NS-2 reste plus attractif car il est libre et gratuit.

Network Simulator NS-2

Network Simulator NS-2 est l'un des outils de simulation des plus populaires au sein de la communauté scientifique. Développé par le département des techniques informatiques à l'université de Berkeley en Californie, NS-2 offre un moteur de simulation de réseaux pour permettre à l'utilisateur de décrire et simuler des communications entre noeuds des réseaux IP. Le simulateur fonctionne avec des scripts écrits en Tcl/Tk ou en OTcl. Une utilisation du C++ est toutefois conseillée pour les simulations délicates, où NS-2 a l'avantage de proposer de nombreuses extensions pour améliorer ses caractéristiques de base. Il est donc possible de modéliser tout type de réseau et de décrire les conditions de simulation: la topologie du réseau (LAN, sans-fil, ..), les caractéristiques des liens physiques, le type de trafic qui circule, les routeurs et les mécanismes d'ordonnancement à appliquer, les protocoles utilisés, les communications qui ont lieu. Il a pour point fort de pouvoir intégrer de nouvelles fonctionnalités et mettre à jour sa bibliothèque : chaque année, une nouvelle version de NS-2 apparaît. NS-2 est organisé sous forme de classes. Il offre une large bibliothèque de modules écrite dans les langages OTcl et C++, caractérisant chaque élément de fonctionnement du réseau. On trouve, par exemple, les classes Node pour définir un noeud, Link pour le lien physique, DropTail pour la politique d'ordonnancement FIFO,... (Fig. 2.9). Ces classes disposent de méthodes qui permettent de personnaliser les propriétés des objets. L'utilisateur peut modifier ces modules et les adapter

à ses propres besoins. Il peut de même en intégrer de nouveaux.

En effet, dans certaines versions de NS-2, il est possible d'implémenter la bibliothèque MNS [70] qui permet de réaliser des simulations basées sur le protocole MPLS. Cette bibliothèque permet donc d'établir des chemins explicites entre une paire de noeuds source/destination en s'appuyant sur le protocole de distribution d'étiquettes CR-LDP (Constraint-Based Routing LDP) [52]. Cette bibliothèque est nécessaire pour l'implémentation de deux algorithmes LBWDP et PEMS proposés pour faire du routage explicite entre une paire de noeuds source/destination.

En plus, NS-2 possède des outils d'animation comme l'animateur de Réseau (Network Animator (nam)) [73],qui est un outil permettant d'analyser ces fichiers. Son interface graphique permet d'afficher la topologie configurée et de visualiser les échanges de paquets, tout comme l'outil xgraph qui est un outil de tracé graphique. Finalement, NS-2 est réputé être un outil permettant d'expérimenter de nouvelles idées et de nouveaux protocoles.

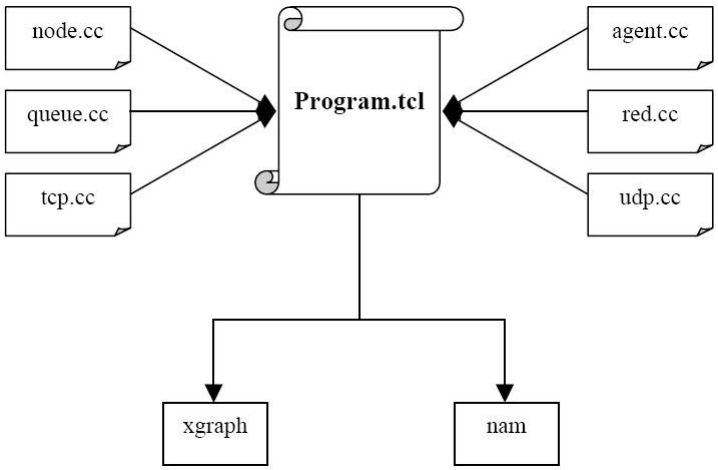


FIG. 2.9 – *Répresentation modulaire de l'outil de simulation NS-2*

2.5.2 Modèle

Dans cette section, nous présentons l'architecture du réseau sur lequel nous avons effectué les simulations. La simulation avec NS-2 passe en général par trois phases: la génération de la topologie, la spécification du scénario du trafic et enfin l'implémentation de l'algorithme et la génération des paramètres de qualité de service. Pour chaque phase, on établit un script Tcl qui intègre les différentes fonctions utilisées pour générer cette phase.

Etape 1: Génération de la topologie

Dans cette étape, on définit la topologie du réseau générée par le générateur automatique de topologies BRITE [13]. Cette topologie contient la structure réseau sur laquelle l'algorithme sera implémenté. Par contre, comme la simulation sous NS-2 est basée sur des scripts TCL, et BRITE, quant à lui, génère des topologies en format "brite", donc afin d'adapter les topologies générées par BRITE, nous avons développé un utilitaire, appelé *brite2ns*, qui permet de transformer une topologie créée par BRITE dans le format TCL adéquat utilisé par le simulateur et directement utilisé par l'algorithme à simuler. Cette topologie contient des noeuds (noeuds IP et noeuds MPLS) et des connexions entre noeuds. Pour chaque lien, on définit le délai, la bande passante et le type de file d'attente se trouvent à l'extrémité. Enfin, on configure les différentes applications MPLS comme le protocole de routage explicite et le protocole de signalisation (CR-LDP). Finalement, on connecte plusieurs paires de noeuds source/destination aux routeurs MPLS.

Etape 2: Spécification du scénario du trafic

Dans cette étape, l'utilisateur spécifie les différents agents de communication qui vont agir pendant la simulation. Un agent de communication consiste

à affecter une source de trafic à un noeud. Il décrit les différentes opérations (envoi de données, volume des demandes, distribution du trafic..). Pour nos simulations, nous avons choisi d'étudier le cas de trois noeuds sources émetteurs. A chacune de ces sources est attribué un noeud destination récepteur. Nous avons choisi d'acheminer un trafic UDP généré aléatoirement, de manière à avoir un scénario de simulation proche de la réalité. C'est-à-dire que plusieurs utilisateurs émettent des requêtes d'applications temps réel sur des intervalles aléatoires. Les paramètres de trafic sont générés selon chaque scénario de trafic présenté dans les sections suivantes.

Etape 3: Implémentation de l'algorithme et génération des paramètres de QdS

Dans cette étape, nous implémentons les fonctions de l'algorithme utilisé pour la réalisation des simulations. L'algorithme ainsi généré se base aussi sur les deux fichiers précédemment décrits (topologie et trafic). Au cours de la simulation, les résultats des paramètres de qualité de service visés sont stockés dans des fichiers. Les principales données archivées sont: l'utilisation moyenne et maximale, le délai moyen de bout en bout. Ces fichiers résultats peuvent être directement exploités par NS-2 ou par l'un des outils qui l'accompagnent (outil de tracé graphique: xgraph, outil d'animation de la simulation : nam). Ils peuvent également être importés par d'autres outils permettant de réaliser des analyses statistiques.

2.6 Vérification du passage à l'echelle par simulation

Un des critères importants de l'évaluation de performances d'une approche dans son environnement d'exécution est sa capacité à résister au facteur d'échelle. Le problème du passage à l'échelle a été bien analysé surtout dans les domaines du calcul parallèle et des systèmes distribués. Ces études font références à des architectures génériques ou des services spécifiques comme le web, où plusieurs métriques composites ont été développées pour capturer leurs passages à l'échelle. Cependant, dans le domaine des réseaux, le passage à l'échelle a été analysé comme un problème de performance où des métriques assez diverses ont été envisagées. Ces métriques se concentrent autour de la mesure des temps de réponse, de la consommation de ressources et bien d'autres métriques qui agissant sur les performances du système. Pour sa définition la plus générale, le passage à l'échelle est: *l'aptitude d'une solution à un problème de fonctionner même si la taille du problème augmente*. En effet, cette définition est insuffisante, puisque le passage à l'échelle ne signifie pas seulement que le système fonctionne, mais qu'il doit améliorer son fonctionnement avec un certain niveau de qualité.

2.6.1 Evaluation de la complexité des algorithmes

Dans notre cas, le passage à l'échelle dépend de deux facteurs : l'étendue du déploiement et la complexité des algorithmiques mettant en oeuvre le modèle. Sur le plan du déploiement, LBWDP et PEMS réduisent le phénomène de taille car ils nécessitent une implémentation complète que sur les routeurs d'entrée. En effet, dans le cas de LBWDP [59], une fois le chemin sélectionné dans le routeur d'entrée, le rôle des routeurs du coeur du réseau se limite

à la mise en oeuvre des mécanismes de MPLS. Dans le cas de PEMS [58], les routeurs du coeur du réseau doivent en plus de ces mécanismes mettre en oeuvre la différentiation des paquets de Diffserv [25]. Comme le passage à l'échelle de Diffserv a été prouvé, cela ne remet pas en cause a priori le fonctionnement de PEMS. Le passage à l'échelle de nos modèles est donc fonction de la complexité des algorithmes de sélection des chemins et de la répartition de charge mise en oeuvre sur les routeurs d'entrée. La mesure de la complexité d'un modèle permet donc de préciser sa capacité de se déployer sur un réseau complexe par sa taille comme internet. Plus la complexité du modèle est faible, plus son implémentation réduit l'utilisation des ressources du réseau. Cette complexité peut s'exprimer selon deux critères : le temps de calcul et l'espace mémoire utilisé. Dans cette étude, on s'intéresse au temps de calcul pour estimer les plans de routage après chaque mise à jour. Ce temps de calcul peut être approximé théoriquement par le nombre d'itérations dans un algorithme. La figure 2.10 montre la complexité théorique de LBWDP et PEMS selon les étapes qui les constituent. On remarque que les étapes les plus coûteuses en temps sont celles relatives à la sélection des chemins.

Algorithmes	Chemins candidats		Répartition du trafic	
	Algorithme	Complexité	Algorithme	Complexité
LBWDP	WDP	$O(n^2)$	PER	$O(n)$
PEMS	Sélection de chemins candidats	$O(n^2)$	PER	$O(n)$

FIG. 2.10 – *Mesure de la complexité des algorithmes LBWDP et PEMS*

2.6.2 Scalabilité et passage à l'échelle des algorithmes

Dans le domaine des réseaux, le facteur d'échelle le plus considéré, pour évaluer la scalabilité d'un algorithme, est le nombre de noeuds. Ce passage à l'échelle de l'algorithme est évalué à grande échelle ou à petite échelle. Un algorithme est scalable à grande échelle, s'il maintient ses performances une fois implémenté sur un réseau d'une taille plus grande. Cependant, il est scalable à petite échelle, s'il maintient ses performances après une limitation de la taille du réseau. Dans cette étude, nous nous intéressons à l'aspect grande échelle, en tenant compte des performances acquises par l'aspect petite échelle. Nous procédons à des simulations avec de topologies de petites et grandes tailles, générées par le générateur automatique de topologies BRITE [13], pour évaluer l'impact de la densité du réseau sur les performances des deux algorithmes LBWDP et PEMS. Rappelons nous que BRITE permet de générer des topologies selon plusieurs structures (AS, routeur, hiérarchique) et selon plusieurs méthodes d'interconnexion entre les noeuds du réseau (Waxman, Barabt').

Les simulations composées de trois scripts Tcl, à savoir la topologie, le trafic et l'algorithme, sont générées de la même manière, en changeant le nombre de noeuds d'un scénario à un autre. Les topologies sont des topologies routeurs, avec la méthode d'interconnexion entre noeuds "Waxman". Les paramètres des topologies utilisées sont indiqués dans la figure 2.11. Comme décrit dans le paragraphe précédent, nous avons choisi d'acheminer un trafic temps réel. La figure 2.12 résume les valeurs des paramètres du trafic utilisées dans les simulations. Les figures 2.13, 2.14, 2.15 montrent respectivement les profils des classes de trafic EF, AF et BE en fonction de temps. Les deux algorithmes LBWDP et PEMS sont simulés à chaque fois avec le même scénario afin de permettre une comparaison objective. Rappelons les

Niveau de topologie	BRITE	Placement nœuds
	Niveau Routeur	Heavy Tailed
Degré des nœuds	Degré minimal est 2	
Bande passante des liens	1.55 [Mbps]	
Délai des liens	10 [ms]	
Type de files d'attente	CBQ	
Max. LSP hop counts	15	
Source/destination	3 paires	

FIG. 2.11 – *Paramètres des topologies*

Temps des simulations	100 [sec]
Demandes du trafic	500 [KB]
Distribution du trafic	Uniforme
Temps de mise à jour	Chaque 3 [sec]
Période des demandes	Une demande toutes les 2 secondes

FIG. 2.12 – *Paramètres du trafic*

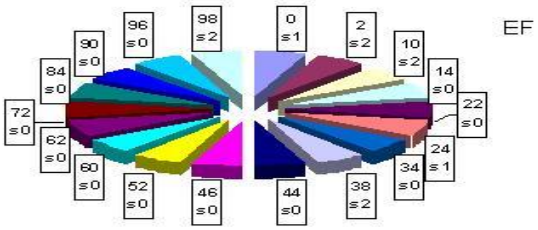


FIG. 2.13 – *Profile du trafic EF en Temps/Source émetteur*

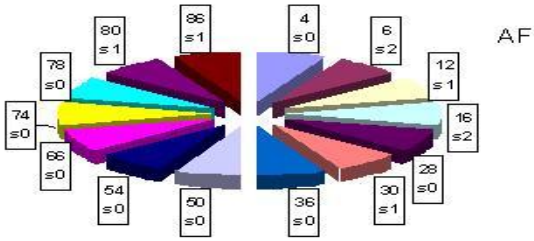


FIG. 2.14 – *Profile du trafic AF en Temps/Source émetteur*

principaux critères pour l'évaluation de deux modèles: l'utilisation moyenne et maximale, et le délai moyen de bout en bout.

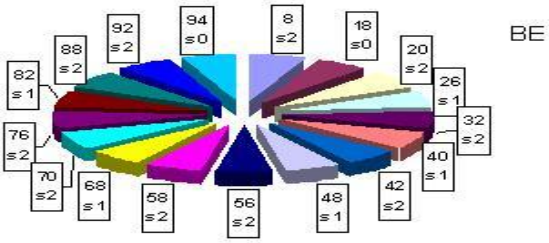


FIG. 2.15 – *Profile du trafic BE en Temps/Source émetteur*

2.6.2.1 Réseau de petite taille

Pour cette première série de simulations, nous avons simulé les algorithmes avec un nombre croissant de noeuds sur petite échelle de 10 à 29 noeuds, afin d'évaluer leurs performances sur des réseaux de petite taille, et pouvoir comparer les résultats avec des simulations sur grande échelle. La figure 2.16 montre l'exemple d'une topologie de 26 noeuds. Les résultats pré-

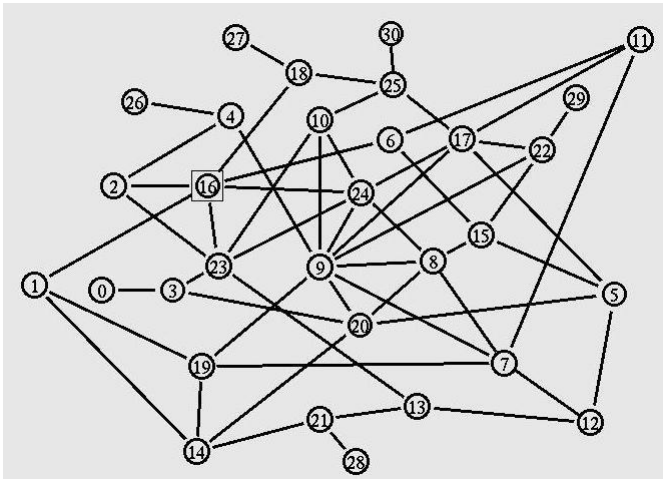


FIG. 2.16 – *Exemple d'une topologie de 26 noeuds générée par BRITE*

sentés par la figure 2.17 montrent que le taux d'utilisation moyen de LBWDP est inférieur à celui de PEMS pour les 5 topologies simulées. Ce qui permet de conclure que LBWDP équilibre mieux la charge du réseau que PEMS. La figure 2.18 montre le taux d'utilisation maximal de deux algorithmes. On remarque que LBWDP possède un taux d'utilisation maximal inférieur à celui de PEMS dans les 5 cas. On en conclut donc que PEMS porte un risque de congestion plus élevé dans ce scénario.

Par rapport à la différenciation de délai entre les classes d'applications, et comme le montre la figure 2.19 , le délai moyen pour PEMS de la classe EF est inférieur à celui des autres classes, ce qui implique que PEMS a différencié

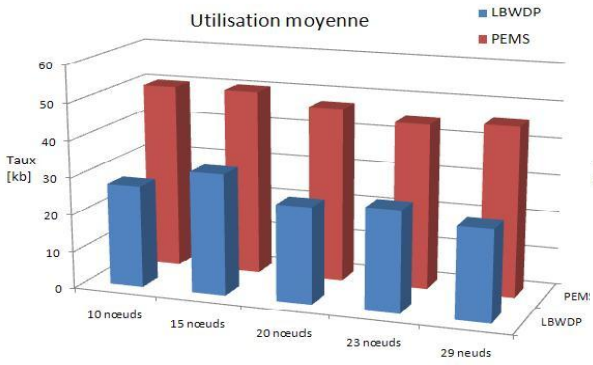


FIG. 2.17 – *Taux d'utilisation moyen pour des topologies de petite taille*

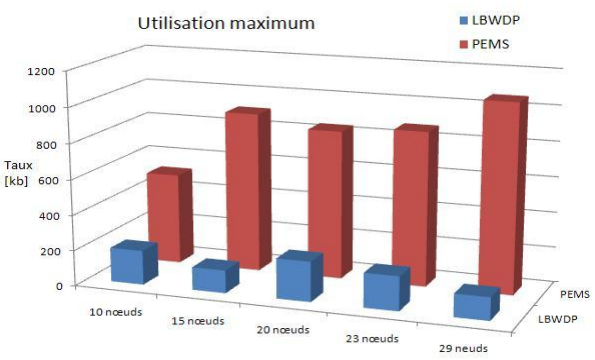


FIG. 2.18 – *Taux d'utilisation maximal pour des topologies de petite taille*

le délai selon les classes de trafic, alors que LBWDP (Fig. 2.20) traite toutes les classes de trafic avec la même priorité. Cependant, la différenciation de délai de PEMS (figure 2.19) reste faible surtout entre les deux classes EF et AF. Par contre, elle parait nettement meilleure pour ces deux classes que pour la classe BE.

2.6.2.2 Réseau de grande taille

L'objectif étant d'étudier la capacité d'adaptation de passage à l'échelle des deux modèles. Ce scénario consiste à mettre en jeu la taille du réseau et la capacité de passage à l'échelle de deux algorithmes. Nous avons choisi d'augmenter considérablement la taille du réseau en simulant une série de topologies de 100 à 300 noeuds.

Les résultats de ce scénario viennent valider ceux obtenus dans le paragraphe précédent confirmant ainsi le comportement déjà observé de deux modèles. Comme on pouvait s'y attendre, LBWDP (Fig. 2.21) possède toujours un taux d'utilisation moyen inférieur à celui de PEMS dans les 3 cas simulées. Donc, LBWDP équilibre mieux la charge du réseau que PEMS dans

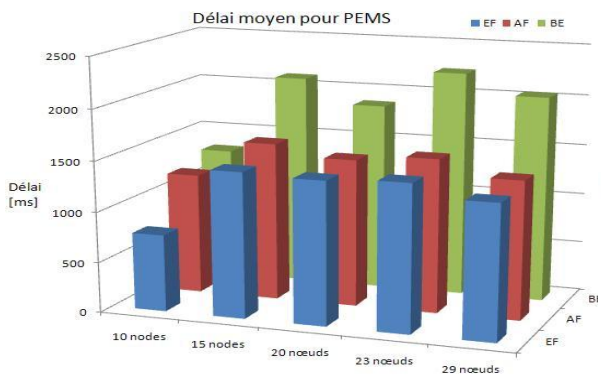


FIG. 2.19 – *Délai moyen de PEMS pour des topologies de petite taille*

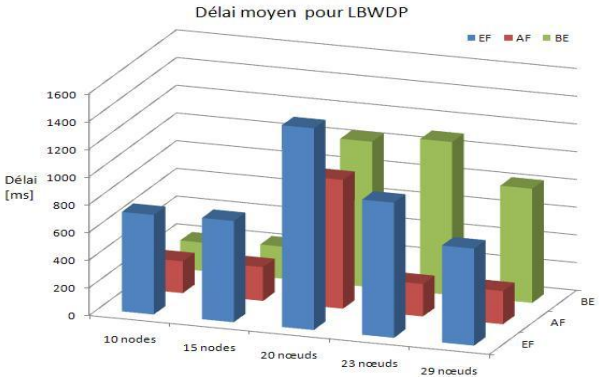


FIG. 2.20 – *Délai moyen de LBWDP pour des topologies de petite taille*

ce cas, car il minimise plus l'utilisation des liens du réseau. Cette conclusion est d'ailleurs renforcée par l'analyse des résultats du taux d'utilisation maximal des liens (Fig. 2.22), où il apparait clairement le risque de congestion provoqué par PEMS.

Les figures 2.23 et 2.24 représentent respectivement les mesures de délai moyen de PEMS et LBWDP. On remarque que PEMS (Fig. 2.23) différencie le délai selon chaque classe de trafic, et il affecte le meilleur délai à la classe EF. Tandis que LBWDP (Fig. 2.24) traite toutes les classes de trafic avec la même priorité, et il n'a pas privilégié l'envoi du trafic premium (classe EF) sur les chemins du meilleur délai. Par contre, l'écart de la différenciation de délai entre les différentes classes de PEMS est bien clair dans ce scénario. Nous pouvons remarquer aussi une stabilité dans les échelles de PEMS contre une perturbation dans les échelles de LBWDP. Cette perturbation dans le cas de LBWDP est due aux fonctionnalités du modèle, qui traitent indifféremment les différentes classes du trafic, et donc le délai varie selon la cadence d'envoi de trafic de chaque classe, de sa quantité ou même de la limitation de la bande passante qui reste un facteur majeur dans la mesure des performances

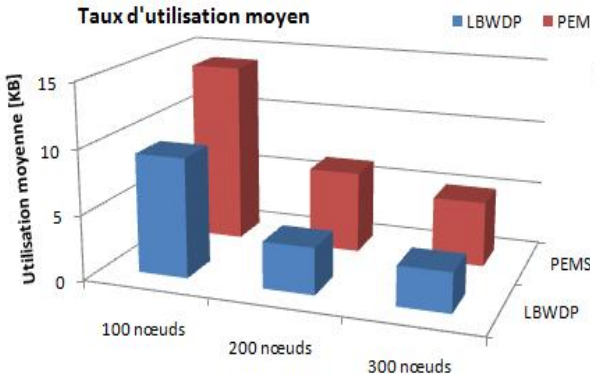


FIG. 2.21 – *Taux d'utilisation moyen pour des topologies de grande taille*

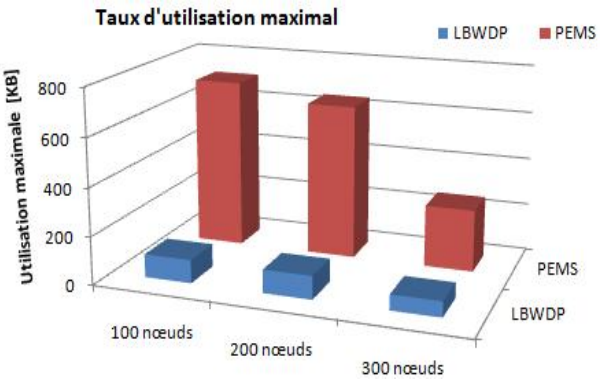


FIG. 2.22 – *Taux d'utilisation maximal pour des topologies de grande taille*

de ces modèles.

2.6.2.3 Conclusion

Les simulations des topologies de petites et grandes tailles permettent d'analyser l'impact de la densité du réseau sur la qualité de l'équilibrage et dans une moindre mesure sur les délais. Concernant l'équilibrage de charge, on constate d'après les simulations que plus le réseau compte de noeuds, mieux la charge est équilibrée. Le nombre de noeuds est une caractéristique de la densité du réseau. Cela signifie donc que l'augmentation du nombre de noeuds entraine l'augmentation du nombre de chemins disjoints entre une source et une destination. Par contre, l'impact de la densité sur le délai est moins mesurable ici. En effet, tout dépend du gain relatif de la baisse des congestions par rapport à l'augmentation de la largeur moyenne des chemins. En général, les conclusions auxquelles nous sommes arrivés concernant le comportement de LBWDP et de PEMS restent toujours valables lors du passage à grande échelle. En effet, sur l'ensemble des scénarios effectués, lors d'un changement de topologie (passage à grande échelle), les deux algo-

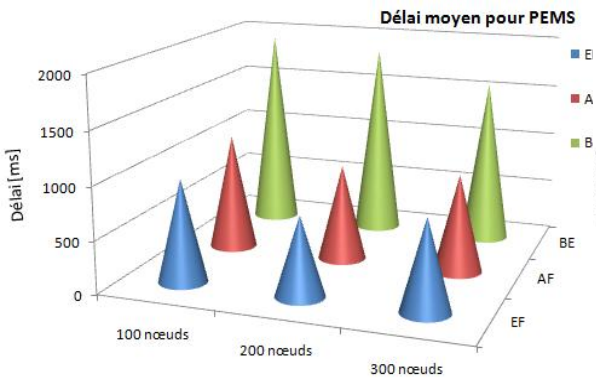


FIG. 2.23 – *Délai moyen de PEMS pour des topologies de grande taille*

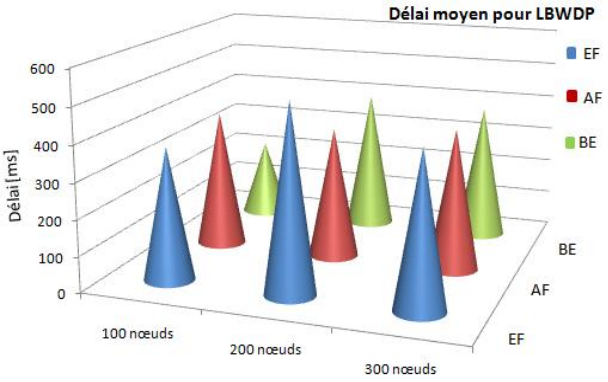


FIG. 2.24 – *Délai moyen de LBWDP pour des topologies de grande taille*

rithmes maintiennent toujours leurs performances et en améliorant certaines. Ces expériences vérifient donc le passage à l'échelle et la scalabilité de ces deux algorithmes.

Par contre, il est nécessaire de signaler que le temps de calcul des chemins, qui est équivalent à l'étape de selection des chemins, augmente considérablement une fois que la taille de la topologie augmente. Mais, cette augmentation n'influence pas les performances de deux modèles, car cette étape s'effectue hors ligne dans la phase de prétraitement.

2.7 Scénarios des simulations pour évaluation qualitative

Dans cette section, nous exposons divers scénarios d'expériences qui représentent l'état de charge d'un réseau FAI selon deux niveaux de charge (trafic faible, trafic élevé), après l'évaluation des algorithmes dans un scénario moyennement chargé dans la section précédente. Les simulations avec un trafic irrégulier permettent d'analyser la capacité d'adaptation des al-

gorithmes à des situations diverses qu'on peut trouver dans un réseau FAI. Nous présentons les résultats par paramètre de qualité de service visé. Autrement dit, nous rappelons que la problématique de cette étude est d'analyser la capacité de l'équilibrage de charge ainsi que la différentiation de service offerte par les deux modèles LBWDP et PEMS.

Dans les deux cas, la topologie du réseau est fixée avant la simulation. Les modèles sont simulés sur une topologie de 100 noeuds générée par BRITE et avec les paramètres indiqués dans la figure 2.11.

2.7.1 Cas 1: réseau faiblement chargé

2.7.1.1 Scénario

L'objectif de ce scénario est d'étudier le comportement des algorithmes LBWDP et PEMS en cas de faible charge et du non risque de congestion du réseau. Le critère de charge du réseau est définit par la capacité résiduelle de ses chemins en terme de bande passante. Plus la valeur de la demande est faible, plus ces chemins seront moins chargés et ainsi le réseau est caractérisé par une faible charge. Les paramètres du trafic sont les mêmes que ceux indiqués par la figure 2.12. La seule variable pouvant être modifiée est la valeur de la demande de trafic envoyée depuis les sources. Cette demande est diminuée et son volume est fixé à 100 [KB].

2.7.1.2 Analyse des résultats

La figure 2.25 montre le taux d'utilisation moyen de LBWPD et PEMS. L'interprétation de la figure 2.25 met en avant l'aspect de l'équilibrage de charge dans le réseau. On remarque que, dans un réseau faiblement chargé, PEMS est capable de fournir un équilibrage de charge équivalent, ou sensi-

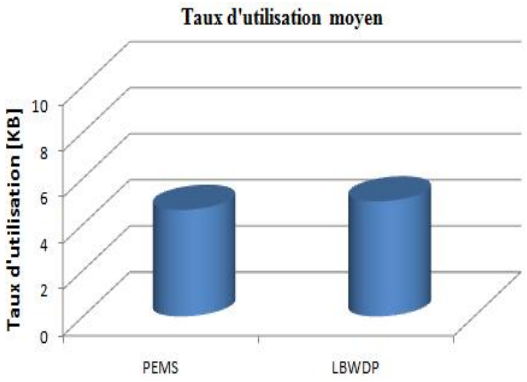


FIG. 2.25 – *Taux d'utilisation moyen pour un réseau faiblement chargé*

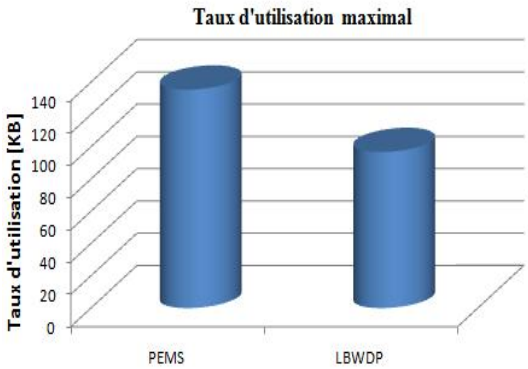


FIG. 2.26 – *Taux d'utilisation maximal pour un réseau faiblement chargé*

blement meilleur, que celui de LBWDP. En effet, la figure 2.25 montre que PEMS possède un taux d'utilisation moyen légèrement meilleur que LBWDP.

Cependant, la figure 2.26 montre que PEMS possède un taux d'utilisation maximal plus élevé que celui de LBWPD. Ce taux maximal est observé seulement à la fin de simulation, où le réseau devient de plus en plus chargé. Par contre, l'écart entre les taux d'utilisation maximaux dans un réseau faiblement chargé (Fig. 2.26) est clairement plus petit que celui dans un réseau moyennement chargé (Fig. 2.22).

La figure 2.27 présente les résultats des délais moyens pour chaque classe de trafic. Celle-ci montre en premier lieu la différenciation visée du point de vue délai de chaque classe par PEMS, à savoir le meilleur délai pour la classe EF par rapport à celui de la classe AF et celui-ci par rapport au délai de la classe BE. Cependant, LBWDP traite toutes les classes de trafic de la même manière.

D'autre part, dans le cas d'une faible charge du réseau, nous remarquons que les délais moyens de PEMS sont constamment inférieurs à ceux de LBWDP (figure 2.27), à l'inverse du cas d'un réseau moyennement chargé

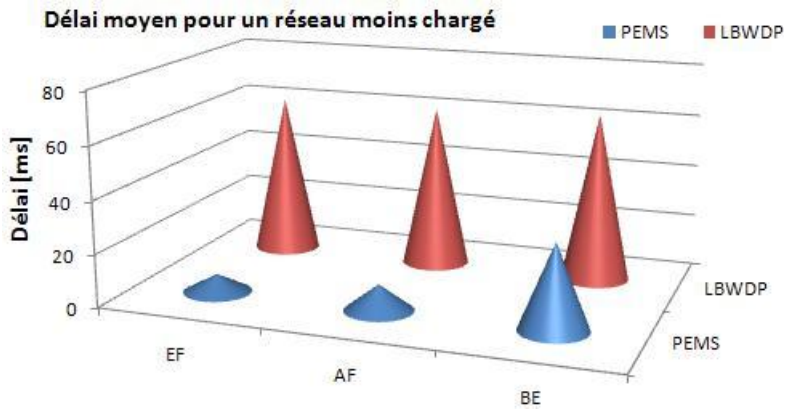


FIG. 2.27 – *Délai moyen pour un réseau faiblement chargé*

(Fig. 2.23 et 2.24). On pense que ceci peut être en lien avec les caractéristiques de chaque modèle, où le critère de la largeur d'un chemin, utilisée par LBWDP, joue un rôle important dans la sélection de bons chemins pour satisfaire une demande de trafic plus élevée que dans le cas d'un réseau non chargé. En effet, dans un réseau non chargé, la largeur d'un chemin n'offre pas cet intérêt majeur, vu que l'utilisation des liens reste limitée. Ainsi les chemins gardent toujours des bandes passantes résiduelles largement suffisantes pour acheminer le trafic sans perturber l'envoi du trafic différencié selon les classes, où le critère du nombre de sauts utilisé par PEMS favorise dans ce cas le délai de l'envoi de ce trafic.

2.7.2 Cas 2: réseau fortement chargé

2.7.2.1 Scénario

Le trafic lié à ce scénario devient plus important, ce qui conduit à surcharger les liens du réseau et se rapprocher de la saturation. Ainsi les chemins utilisés seront de plus en plus chargés. Nous gardons la même topologie de 100 noeuds que dans le cas d'un réseau à faible charge. Les paramètres de

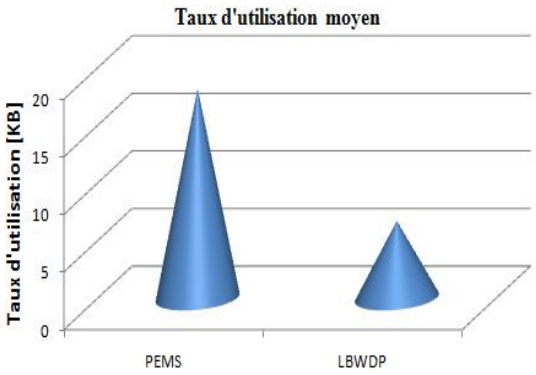


FIG. 2.28 – *Taux d'utilisation moyen pour un réseau fortement chargé*

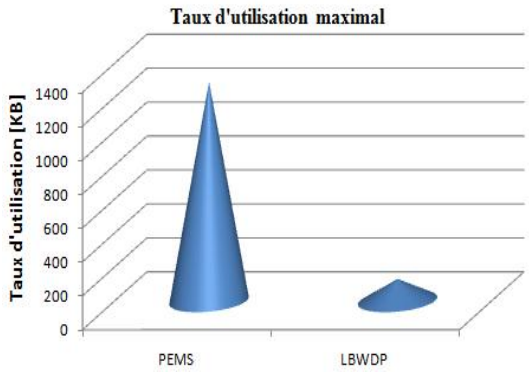


FIG. 2.29 – *Taux d'utilisation moyen pour un réseau fortement chargé*

trafic sont les mêmes indiqués dans la figure 2.12, sauf le volume de demande qui est fixé à 1000 [KB].

2.7.2.2 Analyse des résultats

Dans le cas où la charge du réseau est élevée, la tendance des résultats de l'équilibrage de charge obtenus (Fig. 2.28) est identique au cas d'un réseau moyennement chargé (Fig. 2.21). Les résultats de cette expérience montrent clairement que LBWDP équilibre mieux la charge du réseau que PEMS, car il possède un taux d'utilisation plus faible que celui de PEMS.

Par contre, la figure 2.29 montre que le risque de congestion est bien élevée avec PEMS, car les liens se rapprochent beaucoup de la saturation. Enfin, en comparant les échelles de ces graphes, on remarque que l'écart entre les taux d'utilisation moyens et maximaux augmente de plus en plus avec l'augmentation de la charge du réseau, surtout dans le cas de PEMS, ce qui reflète l'idée que PEMS est plus sensible à l'état de charge du réseau.

De la figure 2.30, on peut constater facilement la différenciation de délai avec PEMS, tandis que LBWDP traite toutes les classes de la même manière.

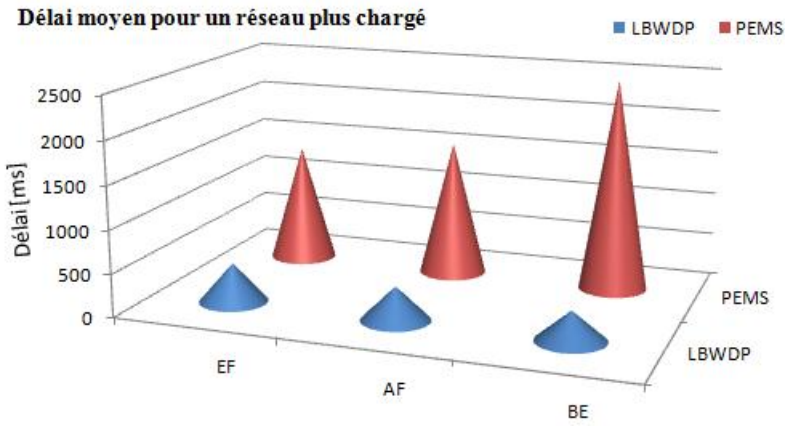


FIG. 2.30 – *Délai moyen pour un réseau fortement chargé*

Finalement, on peut constater que la tendance des résultats obtenus avec un réseau fortement chargé est conforme avec ceux obtenus dans le cas d'un réseau moyennement chargé.

2.7.3 Conclusion

Nous avons présenté au cours de cette section, les simulations de deux algorithmes LBWDP et PEMS avec divers scénarios représentant l'état de charge d'un réseau FAI. Nous avons fixé le nombre de noeuds à 100, en changeant le volume de la demande du trafic selon deux niveaux de charge du réseau: faible et élevée.

Dans un premier temps, nous avons analysé la capacité de deux algorithmes à équilibrer la charge du réseau et à différencier le délai selon les 3 classes de trafic EF, AF et BE dans les deux scénarios. Dans un second temps, nous avons suivi l'effet de l'état de charge du réseau sur les performances des algorithmes, en observant l'équilibre et la corrélation entre les deux principaux critères d'évaluation : le taux d'utilisation moyen et le délai moyen. En effet, nous avons remarqué que l'algorithme qui présente un taux

d'utilisation moyen plus élevé que son concurrent, présente un délai moyen plus important que celui-ci. Cependant, les résultats ont montré que PEMS fonctionne mieux avec un réseau non chargé ou à faible charge, tandis que LBWDP offre un comportement presque similaire avec les deux scénarios et il semble moins influencé par le changement de l'état de charge du réseau.

2.8 Réseau à topologie variante

2.8.1 Introduction

Dans cette partie, nous évaluons la structure des différentes topologies générées par BRITE ainsi que leurs effets sur les performances des algorithmes tout en changeant de méthode de génération des topologies. Les différentes méthodes analysées sont l'emplacement des noeuds (Heavy-tailed, Random) [Fig. 2.31] et l'interconnexion entre les noeuds (Waxman, Barabt'). Les autres paramètres des topologies et de trafic sont illustrés respectivement par les figures 2.11 et 2.12.

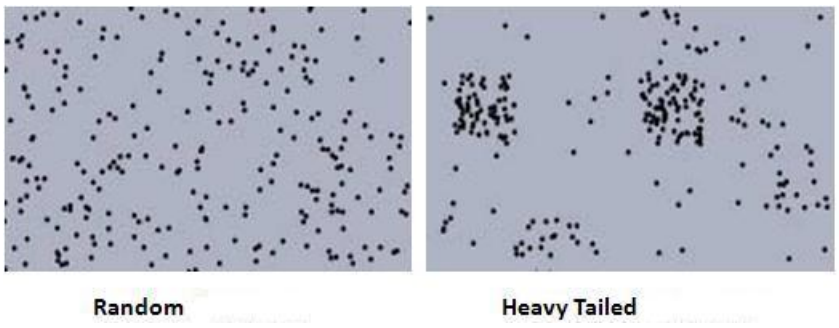


FIG. 2.31 – *Méthodes d'emplacement Random et Heavy-Tailed*

2.8.2 Changement de méthode d'emplacement des noeuds

La figure 2.32 montre le taux d'utilisation maximal de PEMS simulé avec différentes topologies de 100 noeuds générées par les deux méthodes d'emplacement des noeuds (Heavy Tailed : HT-1, HT-2 et Random: R-1, R-2) et avec la même méthode d'interconnexion entre noeuds (Waxman). On peut observer qu'on obtient de meilleurs résultats dans le cas des topologies générées avec la méthode Random que dans celui des topologies générées avec la méthode Heavy-Tailed. Par contre, avec deux topologies générées par la même méthode d'emplacement de noeuds, la différence est moins importante.

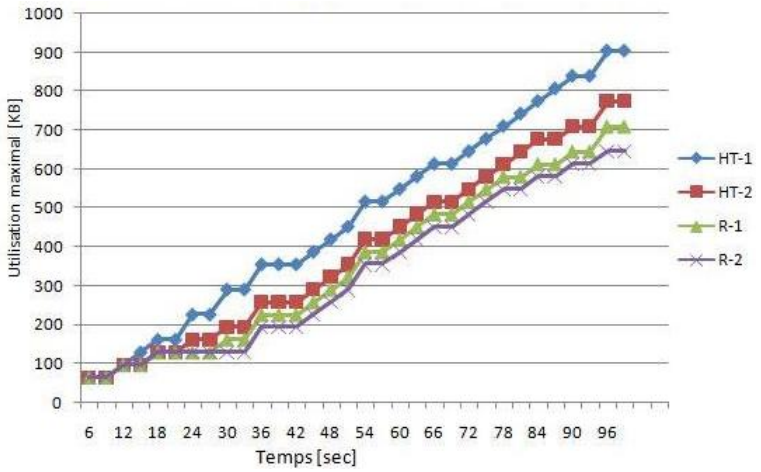


FIG. 2.32 – *Comparaison du taux d'utilisation maximal de PEMS avec les méthodes Random et Heavy Tailed*

2.8.3 Changement de méthode d'emplacement des noeuds et de leur interconnexion

En changeant la méthode d'interconnexion entre noeuds (Waxman, Barabt') et de méthode d'emplacement de noeuds (Heavy-Tailed, Random), on

peut remarquer, d'après la figure 2.33, qu'avec la méthode Barabt' d'interconnexion entre noeuds, le modèle semble plus performant avec les méthodes d'emplacement de noeuds Heavy-Tailed et Random. Avec la méthode Barabt', on obtient à peu près le même taux d'utilisation maximal pour les 2 méthodes d'emplacement de noeuds (Heavy-Tailed, Random). Ce taux est meilleur qu'avec la méthode Waxman d'interconnexion entre noeuds. Ces simulations prouvent que la structure du réseau influence les performances de l'algorithme implémenté.

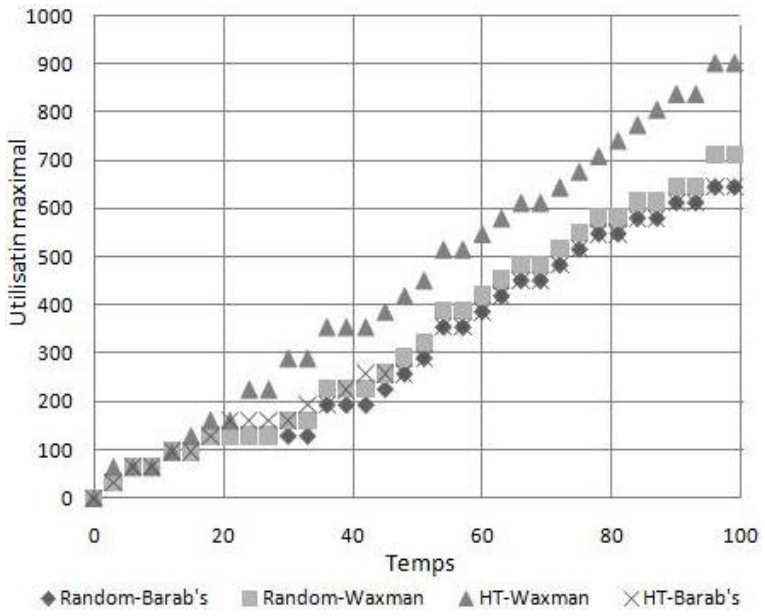


FIG. 2.33 – *Taux d'utilisation de PEMS avec les méthodes d'emplacement et d'interconnexion pour des topologies de 100 noeuds*

Concernant le délai moyen, la figure 2.34 montre que PEMS garde toujours son avantage puisqu'il différencie toujours le délai selon chaque classe de trafic pour toutes les topologies simulées. On peut remarquer aussi que le délai moyen change légèrement selon le type de topologie simulée. Ce qui est cohérent avec le taux d'utilisation obtenu avec la même topologie (figure

2.33). Cette interprétation renforce d'ailleurs l'idée de la corrélation entre le délai moyen et le taux d'utilisation moyen de liens.

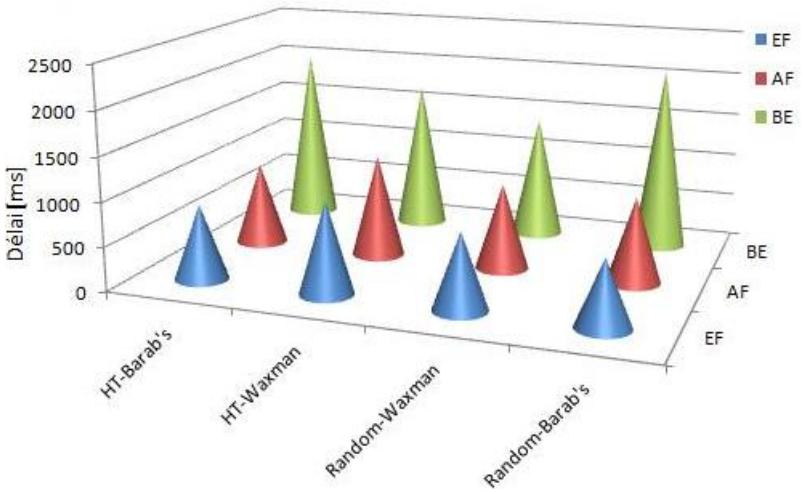


FIG. 2.34 – *Délai moyen de PEMS avec les méthodes d'emplacement et d'interconnexion pour des topologies de 100 noeuds*

2.9 Conclusion

Dans ce chapitre, nous avons évalué les deux modèles LBWDP et PEMS de routage multi-chemins, dans le cadre des technologies MPLS-TE. Leur objectif est de réduire les phénomènes de congestion en équilibrant mieux la charge sur l'ensemble d'un réseau de fournisseur d'accès internet. La technique qu'ils utilisent pour l'ingénierie du trafic est celle du routage multi-chemins sur un réseau IP-MPLS. LBWDP et PEMS reposent sur un premier critère fédérateur pour la qualité de service, à savoir l'utilisation d'un lien. Sur cette base, nous avons comparé ces deux modèles selon différents scénarios pour tester leurs capacités d'adaptation à plusieurs situations. Nous avons simulés les deux modèles sur plusieurs topologies de tailles différentes et en

changeant le volume de trafic. Dans un premier temps, nous avons remarqué que l'équilibrage obtenu dans le cas d'un réseau plus grand est meilleur que celui obtenu dans celui d'un réseau de taille plus petite. Dans un second temps, nous avons constaté que l'état de charge du réseau influence fortement les performances des deux modèles. Leur comportement respectif a changé d'un scénario à l'autre. Enfin, nous avons montré, d'après des simulations sur différentes topologies générées par BRITE, que la structure du réseau agit sur les performances du modèle. Les résultats généraux des différents scénarios sont présentés dans la figure 2.35.

Performance / Trafic	Equilibrage de charge	Différentiation du délai	Comparaison du délai moyen
Trafic élevé	LBWDP équilibre mieux la charge que PEMS	PEMS différencie le délai, LBWDP traite les classes de la même manière	Les délais de LBWDP sont meilleurs que ceux de PEMS
Trafic moyen	LBWDP équilibre mieux la charge que PEMS	PEMS différencie le délai, LBWDP traite les classes de la même manière	Les délais de LBWDP sont meilleurs que ceux de PEMS
Trafic faible	PEMS équilibre sensiblement mieux la charge que LBWDP	PEMS différencie le délai, LBWDP traite les classes de la même manière	Les délais de PEMS sont meilleurs que ceux de LBWDP

FIG. 2.35 – *Résultats d'analyse de performances des deux algorithmes LBWDP et PEMS*

Le changement des performances de deux modèles d'un scénario à un autre, nous amènera dans la suite de cette thèse à développer d'autres techniques liées au comportement de ces algorithmes ou de leurs caractéristiques. Par exemple, nous pourrons mettre en oeuvre la base d'une approche de commande multi-modèles selon l'état de charge du réseau, ou encore nous pourrons développer les bases d'un nouveau modèle de routage multi-chemins, dont lequel le choix des ses critères de sélection de chemins tient en compte

les analyses des résultats obtenus lors de l'étape de l'évaluation de performance des deux modèles LBWDP et PEMS.

Chapitre 3

Modèle dynamique d'état pour le réseau IP/MPLS

L'objectif de ce chapitre est la conception d'un modèle dynamique d'état capable de calculer l'état de charge d'un réseau de type IP/MPLS, et de mesurer la bande passante disponible sur ses chemins de bout en bout. Le but ultime étant d'éviter toute éventuelle situation de congestion dans le réseau, et de proposer dans la suite de cette thèse des approches de contrôle de congestion améliorant ainsi le plan de routage adaptatif et l'équilibrage de charge dans le réseau IP/MPLS. Ce modèle est fondé sur la base de la théorie différentielle du trafic tout en utilisant des approches mathématiques et graphiques.

3.1 Introduction

La complexité et la taille des réseaux de communication ne cessent de croître. Les succès d'Internet et de la téléphonie mobile en sont les exemples les plus spectaculaires. La modélisation permet l'analyse du comportement

de ces réseaux, permettant ainsi une meilleure gestion des ressources.

Notre objectif est de proposer un modèle adaptatif de réseaux à commutation de paquets de type IP, intégrant des mécanismes de différentiation de services et la commutation de labels (étiquettes) MPLS (IP-DiffServ-MPLS), et en particulier le routage explicite dans un domaine IP/MPLS. En effet, la modélisation des réseaux à commutation de paquets et à commutation de circuits a été largement étudiée au cours des trente dernières années. Ces réseaux sont des exemples de systèmes d'une très grande complexité, pour lesquels la modélisation est fréquemment utilisée pour analyser le fonctionnement et étudier les performances.

Au début de ce chapitre, nous faisons un état de l'art concernant les différentes méthodes de modélisation, afin de choisir la plus appropriée pour notre étude et de la valider sur un exemple d'un noeud du réseau. Cette méthode constituera la base du modèle dynamique d'état que nous proposons pour les réseaux IP/MPLS. Nous développons ainsi deux méthodes de représentation basées sur deux approches à compartiments et graphique, afin de proposer une modélisation générale du réseau capable de mesurer dynamiquement son état de charge et calculer les flux de transfert du trafic. Après un exemple d'application, nous introduisons un modèle non linéaire d'état basé sur le modèle proposé caractérisant les phénomènes avec et sans retard.

L'approche adoptée dans ce chapitre consiste à construire un modèle fluide du réseau, basé sur une approximation des équations différentielles exactes gouvernant le trafic moyen transitoire de chaque flot de communication dans le réseau.

3.2 Etat de l'art des modèles

Après un rappel de la notion de modèles, un état de l'art des différents modèles fréquemment utilisés est proposé. Ensuite, nous détaillons la méthode de modélisation retenue.

3.2.1 Notion de modèle

L'objectif principal de ce travail est la conception d'un modèle adaptatif pour les réseaux IP/MPLS. Un tel modèle doit représenter le comportement dynamique du réseau et les phénomènes qui en résultent, spécialement l'état de charge au niveau de ses différents composants (routeurs, liens, chemins..).

Il existe différentes méthodes de modélisation des réseaux. Citons les trois les plus utilisées. La simulation événementielle (OMNeT++, Monte-Carlo) représente l'évolution des systèmes à événements discrets. Cette méthode n'est pas adaptée aux réseaux de grande dimension à cause de son coût calculatoire. La modélisation stochastique (Markov, réseau de Petri..) représente l'évolution du système par une succession d'états discrets. Cette méthode présente un inconvénient lié à l'explosion combinatoire de l'espace d'état pour des réseaux de grande dimension. La méthode qui paraît la plus adaptée est la modélisation d'état. Elle donne une approximation de l'évolution discrète du système par un processus fluide. Son intérêt réside dans la réduction de l'espace d'état par rapport aux deux autres méthodes.

Rappelons qu'un modèle représentant un système de manière précise peut être assez complexe et peu exploitable. Il faut donc trouver un compromis entre précision et complexité.

3.2.2 Etat de l'art

Un modèle de trafic basé sur les files d'attente a été proposé dans [66][65]. Ce modèle est utilisé pour un contrôle de congestion en rétroaction de type "Smith's Principle" dans un réseau ATM caractérisé par la présence du trafic ABR (Available Bit Rate ou Best effort). Le comportement de chaque file est modélisé par intégration numérique des paramètres du taux d'entrée et la somme des différents délais du réseau (propagation, attente, transmission). Le réseau est représenté par un graphe de noeuds et de liens. Le trafic entre une paire source/destination traverse le réseau via un chemin composé de noeuds du réseau. Chaque noeud maintient un buffer pour chaque sortie, et chaque lien a une capacité de transmission $c_i = 1/ti$, avec t_i désignant le temps de transmission d'un paquet, et un délai de propagation noté par T .

La conservation du flux est donnée par l'équation (3.1).

$$x_j(t) = \int_0^t \sum_{i=1}^{n} u_{ij}(\tau - T_{ij})\, d\tau - \int_0^t d_j(\tau)\, d\tau \tag{3.1}$$

Avec:

$x_j(t)$: représente l'état d'occupation de la file associée au noeud j,

n: le nombre de liens partageants la file,

u_{ij}: le taux du flux d'entrée du $ieme$ noeud vers j,

T_{ij}: le délai de propagation du noeud source i vers le noeud destination j (dominant par rapport aux délais de traitement et d'enfilement),

d_j: le taux de paquets sortant du noeud j (ABR available bandwith)

Ce modèle est utilisé pour maintenir le seuil d'occupation de la file d'attente en dessous de sa capacité maximale. Dans [64], ce modèle a été étendu pour une application sur un réseau TCP/IP.

Dans [54], l'auteur propose une approche mathématique pour contrôler

les flux passants par un chemin à travers le réseau. Cette approche a pour but de distribuer, de façon équitable, la bande passante entre les différents utilisateurs du réseau. Elle propose des équations différentielles en fonction du temps, afin d'estimer la charge des chemins du réseau en nombre de flux. L'approche est basée sur les équations suivantes:

$$\frac{dx_r(t)}{dt} = \kappa. \left(\omega_r - x_r(t) \sum_{j \in r} \mu_j(t) \right) \tag{3.2}$$

$$\mu_j(t) = p_j. \left(\sum_{s:j \in s} x_s(t) \right) \tag{3.3}$$

La dérivée par rapport au temps de l'écoulement de trafic $x_r(t)$ est exprimée par la différence entre la demande ω_r et le nombre de paquets marqués en sortie du noeud. En supposant que la ressource j marque une proportion de paquets $P_j(y)$, en insérant un signal feedback quand le flux total traversant j est y, avec w la demande initiale, et k le paramètre du gain supposé positif. Notons que chaque retour est une indication de congestion qui entraine une réduction dans x_r. Cette équation correspond à un contrôle de flux passant par le chemin r, et comprend deux indications: une augmentation du taux correspond à w_r et une diminution du taux correspond à la rafale de signaux de congestion reçus.

Dans [6][5] , les auteurs ont proposé un modèle stochastique à temps discret dont lequel le RTT (Round-Trip delay Time) est considéré comme une unité de temps. Le modèle analyse l'évolution de la longueur de la file d'attente correspondante au lien du goulot d'étranglement. Soit q_n la taille de la file d'attente sur le lien du goulot d' étranglement, μ_n le taux de service correspondant, r_{mn} le taux de trafic entrant depuis la source m, durant la nième période de saut. Alors, l'évolution de la file d'attente est exprimée par

la relation (3.4).

$$q_{n+1} = q_n + \left(\sum_{m=1}^{M} r_{mn} \right) - \mu_n \tag{3.4}$$

Ce modèle a été utilisé pour le contrôle des flux d'entrée ainsi que pour le taux de service, qui peut varier selon les différents types de trafic (CBR, VBR, ABR).

Le modèle retenu se base sur plusieurs critères comme la dynamique, la représentativité, et aussi la facilité à être déployé sur un réseau de type IP/MPLS et ses contraintes de routage et de qualité de service. Ce modèle se base sur la théorie différentielle de trafic présentée ci-dessous, et qui consiste à produire des équations différentielles exactes gouvernant le trafic moyen du système.

3.2.3 Dynamique du modèle fluide pour un noeud

3.2.3.1 Cas SISO

Dans ce paragraphe, nous présentons un modèle fluide qui constituera la base du modèle dynamique d'état pour le réseau MPLS. Ce modèle a fait l'objet de plusieurs études sur la modélisation des réseaux à commutations de paquets et à commutations de circuits [9][14][38]. Il est basé sur la théorie différentielle de trafic. Il a été développé par Garcia et Legall [34] [35][36][37]. Il permet l'étude du trafic au niveau de la file d'attente du routeur d'un réseau d'une manière analytique. En effet, dans les travaux liés à cette théorie [14][36][38], les auteurs définissent la théorie différentielle du trafic comme une méthodologie de modélisation analytique fluide qui permet l'étude du trafic des ressources d'un réseau aussi bien en régime stationnaire qu'en régime transitoire. Le modèle est présenté dans [36] de la manière suivante:

Définissons le système élémentaire d'un noeud contenant une file d'attente

de type M/M/1/∞ représenté par la figure 3.1:

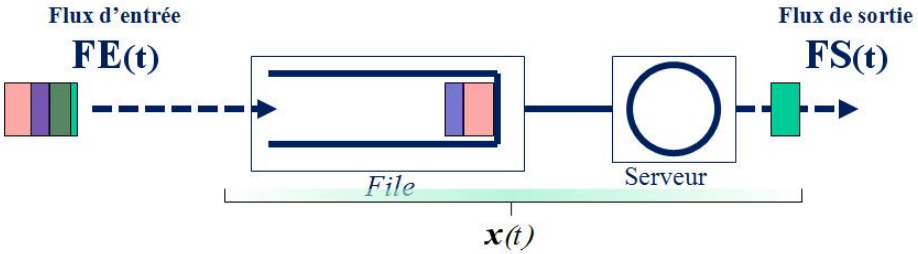


FIG. 3.1 – *Système d'attente élémentaire*

Avec $x(t)$ est le nombre de paquets dans le système, et $\dot{x}(t)$ sa dérivée. Soient FE(t) et FS(t) respectivement les débits instantanés d'entrée et de sortie du système élémentaire. Le trafic moyen transitoire est alors donné par la différence entre les flux instantanés d'entrée FE(t) et de sortie FS(t) selon l'équation différentielle suivante:

$$\dot{x}(t) = \textbf{\textit{Flux d'entrée - Flux de sortie}} = \textbf{\textit{FE(t) - FS(t)}}$$

Dans [36], l'auteur exprime que l'idée fondamentale de la théorie différentielle du trafic est d'établir l'expression exacte de l'équation différentielle gouvernant le trafic moyen transitoire. Cette expression est obtenue par une analyse du processus markovien ou semi-markovien associé à $x(t)$. Sa forme générale est donnée par l'équation (3.5):

$$\dot{x}(t) = FE(t) - FS(t) = \sum_i x_i(t).\frac{dP[x_i(t)]}{dt}, \qquad x(t_0) \quad donné \qquad (3.5)$$

(avec P[x(t)] est la probabilité de l'état x(t))

Supposons maintenant que le flux d'entrée $FE(t)$ est modélisé par le paramètre λ, le flux de sortie $FS(t)$ est représenté par f, et la file d'attente au niveau du noeud possède un taux de service μ fixé par l'administrateur. Alors le système de la figure 3.1 sera représenté par la figure 3.2 :

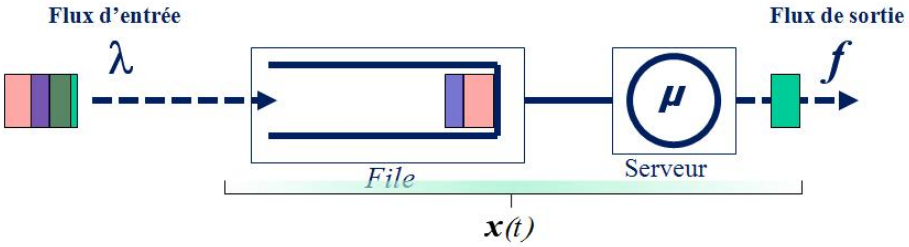


FIG. 3.2 – *Paramétrage d'un système d'attente élémentaire*

La résolution du processus de Markov associé [36][37] conduit à une approximation dynamique qui dépend de l'état interne de la file d'attente de la forme:

$$\dot{x}(t) \cong \lambda - \mu \frac{x(t)}{1 + x(t)}, \qquad x(t_0) \quad donné \tag{3.6}$$

L'intégration numérique de l'équation différentielle (3.6) permet de calculer $x(t)$. Cette équation exprime la charge d'un noeud par la différence entre les flux d'entrée et de sortie et donne une approximation dynamique du nombre de clients (paquets) dans le noeud.

Nous avons vu précédemment les expressions qui régissent le comportement des éléments fondamentaux du modèle fluide à savoir l'état de la file d'attente en nombre de clients, le débit d'entrée et de celui de sortie... En effet, ce modèle est à temps continu qui dispose de composants interagissant par l'intermédiaire de signaux à temps continus. La simulation de modèle continu implique la résolution numérique des ses équations différentielles en utilisant les techniques usuelles d'intégration, l'état du modèle est alors obtenu par calcul dynamique d'un point fixe. Dans un modèle fluide, les paquets sont considérés comme un flot de données continu ou "fluide" qui s'écoulent dans le réseau. Le nombre de changements d'état est ainsi réduit. Cette technique

permet d'améliorer la simulation en la rendant moins coûteuse en nombre d'événements, donc en temps d'exécution et en espace mémoire.

La simulation analytique permet d'illustrer la validité de l'approche fluide du point de vue des résultats obtenus. Nous comparons les résultats de la simulation analytique avec ceux issus des simulations événementielles et des simulations du modèle stochastique de la théorie des files d'attente.

La théorie des files d'attente [19][40] est la plus utilisée dans l'évaluation des systèmes de communications. Elle représente et analyse des systèmes à ressources partagées. Elle permet d'exprimer les valeurs analytiques exactes (taux d'utilisation, nombre de clients, intervalle d'arrivée..) d'un système en traitant une série de probabilités stationnaires. La simulation événementielle (en utilisant dans notre cas le langage Tcl/Tk [89]) reste une estimation qui donne une idée sur l'évolution du système en fonction du temps.

Le tableau (3.3) présente les principaux paramètres d'un noeud élémentaire, contenant une file d'attente de type M/M/1/∞, un taux de service μ, un débit d'entrée λ.

File d'attente	M/M/1 (de grande capacité)
Taux de service μ fixe	7 (flux par période)
Débit d'entrée λ	5 (flux par période)
Temps de simulation	100 secs

FIG. 3.3 – *Les principales valeurs du système élémentaire*

Dans le tableau (3.4), on compare le nombre moyen de clients (colonne 2) avec ceux obtenus par intégration du modèle différentiel (colonne 1) et avec celle obtenue par simulation événementielle (colonne 3). Les résultats montrent la très bonne estimation obtenue avec le modèle différentiel.

Le graphe 3.5 confirme cette conclusion en comparant la résolution du

	Résolution analytique du modèle fluide	Calcul selon la théorie de files d'attente	Simulation événementielle
$x_{(t)}$	6.000	6.000	6.066

FIG. 3.4 – *Comparaison des résultats obtenus par les différentes méthodes*

modèle fluide différentiel et la simulation événementielle. On remarque que le modèle fluide donne une bonne approximation du comportement du noeud, et dans la plupart des cas, il est plus précis que la simulation événementielle.

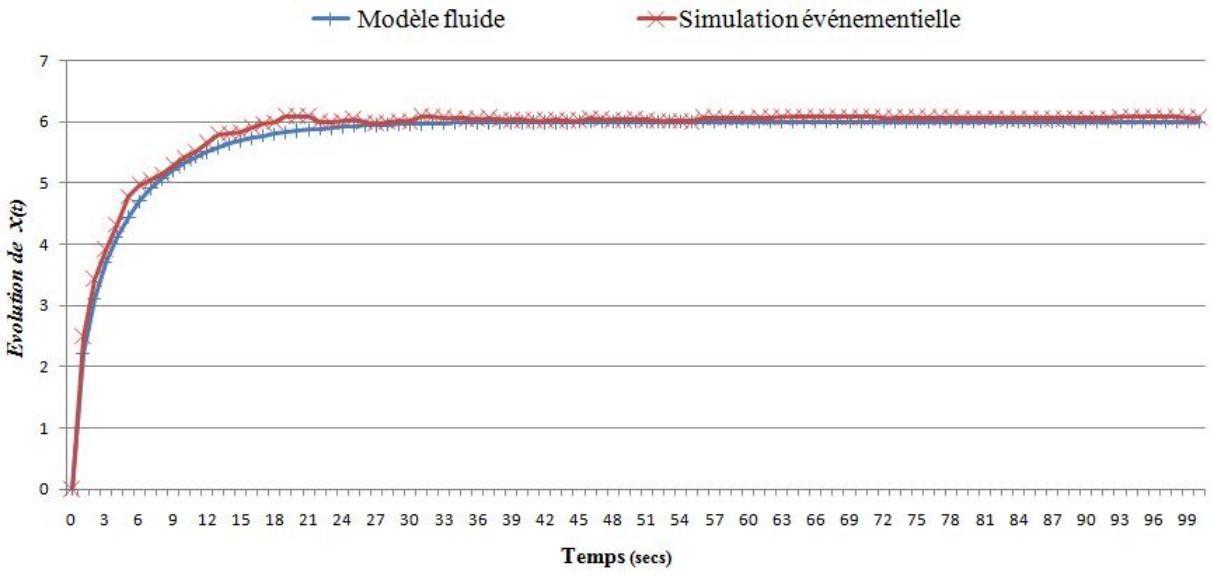


FIG. 3.5 – *Comparaison entre les résultats obtenus par intégration du modéle fluide et par simulation événementielle*

3.2.3.2 Cas MIMO

Un noeud MIMO est composé d'une file d'attente et de plusieurs entrées/sorties (Fig. 3.6). Les entrées peuvent être directes, provenant de noeuds sources du réseau MPLS ou des interactions entre les noeuds internes du réseau. Les sorties sont les interactions entre les noeuds ou des flux livrés aux

destinations correspondantes.

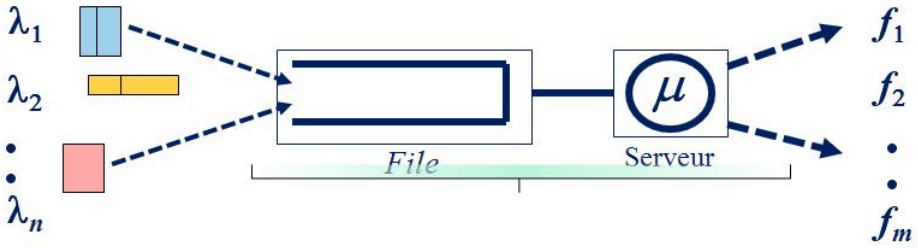


FIG. 3.6 – *Répresentation d'un système d'attente MIMO*

Le modèle fluide appliqué sur un noeud SISO (Single Input Single Output) reste valable pour un système MIMO (Multiple Input Multiple Output). L'équation générale du système MIMO est définie en utilisant le même principe de la théorie différentielle de trafic, par la différence entre *la somme* des débits instantanés d'entrées et ceux des sorties. Supposons qu'on dispose d'un noeud avec n entrées et m sorties. Alors, le modèle fluide sera exprimé par la relation (3.7).

$$\dot{x}(t) = \sum_{i=1}^{n} \lambda_i - \sum_{j=1}^{m} f_j = \sum_{i=1}^{n} \lambda_i - \mu * \frac{x}{1+x} \tag{3.7}$$

La somme des flux de sorties est approximée par la relation: $\mu * \frac{x}{1+x}$

3.3 Modèle dynamique d'état pour un réseau IP/MPLS

Cette section porte sur la modélisation d'un réseau IP/MPLS à l'aide du modèle différentiel fluide. Rappelons qu'un réseau MPLS est capable de fournir des chemins explicites entre une paire de source/destination. Ces chemins sont calculés à l'avance en se basant sur certains critères de qualité

de service comme le délai, la bande passante, et bien d'autres. Pendant la phase de communication, le réseau utilise ces chemins pour acheminer le trafic entre cette paire de source/destination. Cette technique de routage explicite constituera la principale hypothèse lors de l'étape la modélisation du réseau IP/MPLS.

Cependant, l'équation générale du modèle fluide élémentaire (Eq. 3.6), montre que le débit instantané de sortie d'un noeud est directement additionné au débit instantané d'entrée du noeud suivant. Nous estimons donc qu'il est possible d'en tirer partie pour développer les équations différentielles fluides des noeuds interconnectés, ainsi que le système différentiel global du réseau entier.

Cette section est structurée de la façon suivante: on commence par mettre en oeuvre un modèle général qui exprime le mécanisme de transfert du flux dans le réseau. Ensuite, on développe une méthode qui permet de distribuer le flux de sortie d'un noeud vers ses voisins, en fonction de l'information du taux de trafic acheminé sur les chemins du réseau. A ce stade, une connaissance de la topologie du réseau devient indispensable pour modéliser les interactions entre les différents composants du réseau, et ainsi déterminer les noeuds responsables de l'acheminement du trafic sur les chemins du réseau. Pour cela, nous définissons une approche graphique qu'on pourra utiliser pour développer une méthode capable de déterminer ces noeuds, et de modéliser le plan de routage correspondant au processus MPLS.

3.3.1 Approche des réseaux à compartiments

Les réseaux à compartiments [42] ont été utilisés dans de nombreux domaines, comme les réseaux de communications [7][43], les procédés industrielles et économiques [44][47], les systèmes de santé [39], etc. Le principal

intérêt de ces réseaux consiste en l'élaboration d'une représentation dynamique du comportement du système à modéliser. En nous appuyant sur l'approche des réseaux à compartiments, nous présentons dans ce paragraphe une méthode pour la mise en équation d'un modèle dynamique pour le réseau MPLS. Ce réseau est représenté par un graphe unidirectionnel (figure 3.7).

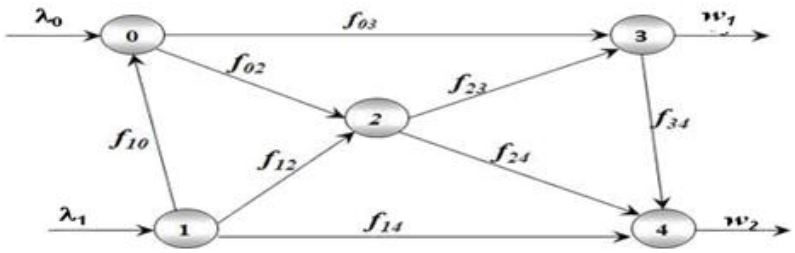


FIG. 3.7 – *Exemple d'un réseau à compartiments*

Les noeuds du graphe, qui représentent les routeurs du réseau, contiennent chacun une quantité variable appelée souvent "compartiment". Cette quantité représente l'état du noeud, dans notre cas, c'est le nombre de paquets dans la file d'attente correspondant au noeud. Les compartiments sont interconnectés par les arcs. Chaque arc constitue un lien entre deux noeuds différents (les liens sont unidirectionnels conformément au réseau MPLS). Les flux de trafic envoyés par les noeuds traversent les arcs (liens) pour aller des sources jusqu'aux destinations des paquets. Le flux de transfert entre deux noeuds i et j est noté par f_{ij}. Les interactions du réseau avec son milieu extérieur (Entrées/Sorties) sont notées par: λ_i envoyées depuis les noeuds sources, et ω_i envoyées vers les noeuds destinations. Soit $x_i(t)$ la quantité variable contenue dans chaque compartiment. Le vecteur $x(t) = (x_1(t), x_2(t), ..., x_n(t))$ est le vecteur d'état du système, son ordre est équivalent à tous les noeuds du réseau MPLS. Les flux de transfert $f_{ij}(x(t))$ entre les noeuds i et

j, et les flux de sortie vers les destinations $w_i(x(t))$, sont des fonctions des variables d'état. Ainsi, la dynamique de chaque compartiment est décrite par l'équation différentielle (3.8).

$$\dot{x}_i = \lambda_i \ + \ \sum_{h \neq i} f_{hi}(x) \ - \ \sum_{k \neq i} f_{ik}(x) \ - \ \omega_i(x), \qquad \dot{x}_i \geq 0 \qquad (3.8)$$

l'évolution de l'état de chaque compartiment est décrite par la différence entre la somme des flux d'entrée f_{hi}, λ_i et celle des flux de sortie f_{ik}, ω_i.

Nous rappelons ci-dessous certaines propriétés structurelles de ces réseaux décrites dans [7][8].

Un réseau à compartiment est un système positif.

Définition 1[7][8]. Un système dynamique $\dot{x} = f(x,t)$, $x \in \mathbb{R}^n$ est positif si: $x(0) \in \mathbb{R}^n_+ \Rightarrow x(t) \in \mathbb{R}^n_+ \qquad \forall\ t \geq 0$.

Propriété 1[7][8]. *Un réseau à compartiments est un système positif.* Le système (4.1) est un système positif. En effet, si $x \in \mathbb{R}^n_+$ et $x_i = 0$ alors $\omega_i(x_i = 0$ et $\dot{x}_i = \sum_{i \neq j} f_{ji}(x(t)) + \lambda_i$ - $\omega_i(x_i) \geq 0$. Ceci est suffisant pour garantir l'invariance de l'ensemble non-négatif si les fonctions $f_{ij}(x)$ et $\omega_i(x)$ sont différentiables.

La quantité totale contenue dans le système est $M(x) = \sum_{i=1}^{n} x_i$. Dans [7][8], les auteurs expriment qu'un réseau à compartiments est conservatif en ce sens que la quantité totale contenue dans le système est conservée.

Propriété 2. *Conservation*[7][8]. Un modèle de réseau à compartiments est dissipatif par rapport au taux d'alimentation (supply rate)$\sum_i \lambda_i(t)$ avec la quantité totale $M(x)$ comme fonction d'accumulation (storage function) dont la dynamique est donnée par l'équation suivante:

$$\frac{dM(x(t))}{dt} = \sum_i \lambda_i(t) \text{ - } \sum_i \omega_i(x(t))$$

En effet, dans [7][8], les auteurs précisent que dans le cas particulier d'un système fermé sans flux d'entrée ($\lambda_i = 0, \forall\ i$), et sans flux de sortie ($\omega_i(x) = 0, \forall\ i$), on vérifie que $dM(x)/dt = 0$, ce qui montre que la quantité totale contenue dans le système est effectivement conservée.

Définition 2. *Réseau connecté aux entrées et aux sorties*[7][8]. Un compartiment i est connecté à une sortie s'il existe un chemin $i \rightarrow j \rightarrow k \rightarrow ... \rightarrow l$ partant de ce compartiment et se terminant en un compartiment l à partir duquel il y a un flux de sortie $\omega_l(x)$. Le réseau est complètement connecté aux sorties (CCS) si chaque compartiment est connecté à une sortie.

Un compartiment l est connecté à une entrée s'il existe un chemin $i \rightarrow j \rightarrow k \rightarrow ... \rightarrow l$ jusqu'à ce compartiment et partant d'un compartiment i dans lequel il y a un flux d'entrée λ_i. Le réseau est complètement connecté aux entrées (CCE) si chaque compartiment est connecté à une entrée.

Le modèle (3.8) apporte une réponse à la question de la mise en oeuvre d'un modèle global pour le réseau de type IP/MPLS. En effet, les routeurs d'un tel réseau seront représentés par les noeuds (compartiments) qui contiennent une file d'attente représentant les réservoirs de stockage et de traitement de la quantité d'information qui circule entre les noeuds, les flux d'entrée et de sortie seront respectivement les λ_i et ω_i, le trafic circulant entre les routeurs sera représenté par les flux de transfert f_{ij}. En supposant que les paramètres de flux d'entrée et du taux de service de files d'attente (μ) sont connus. Il s'agit maintenant de calculer les valeurs de flux de transfert entre les noeuds f_{ij}. Pour cela, on a développé la Méthode d'Assemblage à Compartiments (MAC).

3.3.1.1 Méthode d'Assemblage à compartimants (MAC)

La Méthode d'Assemblage à compartimants (MAC) permet de calculer le flux de transfert entre les noeuds d'un réseau constitué de routeurs (les noeuds) et de liens (les arcs). On suppose que chaque routeur contient une file d'attente qui sert d'espace de stockage et de traitement du flux de paquets. Un routeur peut être connecté à un ou plusieurs routeurs selon la topologie du réseau. Un routeur peut appartenir à plusieurs chemins. Le flux de transfert sortant de ce type de routeur sera distribué sur les différents liens reliant le routeur avec ses voisins. Donc, le paramètre f_{ij}, circulant entre les noeuds i et j.

Le routage est modélisé par un poids u_{ij} appelé "variable de routage", affecté au flux f_{ij}, dans le noeud i, sur le lien le connectant au noeud j. Pour un flux donné, la somme des poids de tous les liens de sortie vaut 1. Le partage des charges sur les chemins est ainsi possible. La variable de routage sera calculée dans la suite de ce chapitre.

$$\sum_{j \neq i} u_{ij} = 1 \tag{3.9}$$

Notons par $r(x_i)$, le flux de paquets sortant du noeud i. Le flux de transfert sur le lien (i,j), $f_{ij}(x_i)$, est exprimé par la relation (3.10).

$$f_{ij}(x_i) = u_{ij} * r(x_i) \tag{3.10}$$

Le coefficient de routage u_{ij} correspond à la proportion du flux $r(x_i)$ partant du noeud i vers le noeud j. En effet, le débit de sortie de ce flux est multiplié par le coefficient u_{ij}, ensuite il sera injecté dans le noeud j.

$r(x_i)$: représente le taux de trafic servi par la file d'attente et qui est supposé une fonction de l'état interne de la file x_i.

Pour calculer le taux de trafic $r(x_i)$, on fait appel au modèle différentiel fluide qui permet d'effectuer une approximation dynamique du flux de sortie

de la file d'attente du noeud i vers le lien de sortie correspondant au noeud j de la façon suivante:

$$r(x_i) = \mu_i \frac{x_i}{1 + x_i} \tag{3.11}$$

En rassemblant les deux équations (3.10) et (3.11), l'équation générale du flux de transfert correspondant au lien (i,j) , $f_{ij}(x_i)$, est donnée par la relation (3.12).

$$f_{ij}(x_i) = u_{ij} \ (\mu_i \frac{x_i}{1 + x_i}) \tag{3.12}$$

3.3.2 Approche graphique

La représentation graphique du réseau suppose une connaissance exacte de la topologie, en particulier, une vue mixte (locale et globale) de l'ensemble des éléments qui entrent en jeu dans le fonctionnement des services fournis par le réseau est nécessaire. Donc, à partir des données disponibles sur la topologie, ainsi que des hypothèses et les services fournis par l'application visée, on va collecter les informations nécessaires pour la génération de la représentation dynamique du réseau.

Considérons à présent le cas des réseaux IP/MPLS, i.e. mettant en oeuvre une technique de routage explicite entre les paires sources/destinations. Dans ce paragraphe, nous mettrons en oeuvre l'approche des réseaux à compartiments ci-dessus, associée à la Méthode d'assemblage à Compartiments (MAC). Dans un réseau IP/MPLS utilisant le routage explicite, seule une partie des noeuds est concernée par l'acheminement du trafic à travers les chemins reliant les différentes paires sources/destinations. La détermination de ce sous-ensemble est indispensable pour le reste de la mise en oeuvre du modèle précis du réseau. En effet, ce sous-ensemble permet de représenter les noeuds émetteurs et récepteurs du trafic à chaque instant. Cette information servira pour le calcul des variables du routage en premier lieu, et aussi pour la

détermination des paramètres effectives des flux de transfert "f" qui seront explicités dans l'équation générale du modèle.

A travers une représentation matricielle, l'approche graphique consiste à établir des relations entre les composants du réseau, en les illustrant par des matrices d'incidence à deux dimensions sous forme de 0 et 1. Si une relation entre deux composants est vérifiée, alors le coefficient correspondant est mis à 1, sinon il sera mis à 0. Ensuite, ces matrices seront insérées dans l'équation générale (3.8), où son développement permet d'obtenir le système d'équations global du système considéré.

3.3.2.1 Méthode de Transition Graphique (MTG)

Afin de modéliser les interactions entre les différents composants du réseau (noeuds, liens, chemins..), nous avons développé la Méthode de Transition Graphique (MTG) qui, en se basant sur l'approche graphique ci-dessus, permet la génération des matrices d'incidenc représentant ces interconnexions. En effet, le contexte de cette étude se base sur le principe du routage explicite qui permet de préciser l'ensemble des chemins possibles entre des paires sources/destinations. A partir de ces chemins, nous allons extraire les sous-ensembles des routeurs et des liens acheminants le trafic dans le réseau. Le sous-ensemble de routeurs constituera le vecteur d'état du système d'équations général du réseau. Les liens seront utilisés pour expliciter les flux de transfert effectifs f(non nuls) entrants et sortants de chaque routeur du sous-ensemble de routeurs concernés par l'acheminement du trafic. Disposant de ces informations, nous allons les écrire sous forme d'un algorithme de transition entre les différentes matrices d'incidences.

- Soit $G = (R, L)$, le graphe orienté modélisant le réseau. Rappelons ici que le graphe est unidirectionnel en corrélation avec le principe de routage expli-

cite de MPLS.

- Soit R, l'ensemble des noeuds du graphe, autrement dit les routeurs du réseau. $R = \chi \bigcup P$, où χ correspond à l'ensemble des noeuds de la périphérie du réseau (les paires sources/destinations) et P est l'ensemble des routeurs du domaine MPLS. Chaque routeur de l'ensemble P est modélisé par une file d'attente de type $M/M/1/\infty$. Soit $x_i(t)$ le nombre de paquets dans la file d'attente associée au routeur i à chaque instant. A partir de l'ensemble P des routeurs du domaine MPLS, nous allons extraire le sous-ensemble T de routeurs concernés par l'acheminement du trafic. Les éléments du sous-ensemble T définissent le vecteur d'état associé au modèle général du réseau. L'ordre du modèle est donc égal à la dimension de T. Le vecteur d'état du système est un vecteur colonne défini par: $x(t) = \{x_i(t),\, i \in T\}$.

- Soit L, l'ensemble des liens du graphe. Ces liens transportent les flux de transfert "f" échangés par les routeurs. A chaque routeur, on trouve des flux entrants des routeurs voisins et des flux sortants vers des routeurs voisins.

- Soit SD, l'ensemble des paires sources/destinations de l'ensemble χ. En accord avec la terminologie du protocole MPLS, chaque paire source/destination est connecté à travers plusieurs chemins explicites. Soit CH l'ensemble des chemins entre les différentes paires sources/destinations. En principe, chaque source envoie un flux de paquets vers la destination en utilisant les chemins établis à l'avance. Soit I l'ensemble du flux envoyé sur les chemins. I est un vecteur colonne défini par: $I = \{I_c,\, c \in CH\}$.

L'algorithme de transition

L'algorithme de transition permet de mettre en oeuvre les relations entre les différents ensembles ci dessus. Il comporte quatre étapes .

- La première étape est celle de l'initialisation, où on construit la matrice

B des relations "sources/destinations—chemins" à partir des informations fournies par le plan de routage explicite. Les éléments en ligne et en colonne sont respectivement composés par l'ensemble des paires sources/destinations et par l'ensemble des chemins.

La matrice B est exprimée par la forme suivante:

$B = \{ B_{sc} , c \in CH \text{ et } s \in SD \}$ avec $B_{sc} = \{$ *1 si s est servi par c et 0 sinon*$\}$

- La deuxième étape consiste à calculer la matrice C des relations "chemins—routeurs". Les termes en ligne sont les chemins possibles et ceux des colonnes sont les noeuds du réseau. La matrice C est exprimée par la forme suivante:

$C = \{C_{cr}, c \in CH, r \in R\}$ avec $C_{cr} = \{$ *1 si r ∈ c et 0 sinon*$\}$

- Dans la troisième étape, et à l'aide des deux étapes 1 et 2, on définit la matrice E des relations "sources/destinations—routeurs". Cette matrice permet de déterminer le sous-ensemble des routeurs concernés par l'acheminement du trafic. Ses lignes se composent des paires sources/destinations et ses colonnes se composent des routeurs du réseau. E est exprimée par la fonction suivante:

$E = B.C = \{ E_{sr} , s \in SD \text{ et } r \in R\}$ avec $E_{sr} = \{$ *1 si s est servi par r et 0 sinon*$\}$

Le sous-ensemble T des routeurs concernés par l'acheminement du trafic est calculé à partir de la matrice E des relations "sources/destinations—routeurs" de la façon suivante:

$$T = \{r, r \in R \ / \ \exists \ s \in SD \text{ et } E_{sr} = 1\}$$

- Enfin, la quatrième étape consiste à calculer les relations des flux de transfert entrants et sortants depuis un routeur. Cette relation sera utilisée pour modéliser les interactions entre les routeurs acheminant le trafic dans le réseau. A partir de la matrice C des relations "chemins—routeurs", on calcule deux matrices de flux entrants G et sortants H. G et H sont deux

matrices dynamiques qui dépendent de la variation de la matrice C, elles sont définies par les expressions suivantes:

$G = \{G_{ji}, i \in R \text{ et } j \in R\}$ avec $G_{ji} = \{$ *1 si (j,i) est un lien entrant à i et 0 sinon*$\}$

$H = \{H_{ik}, i \in R \text{ et } k \in R\}$ avec $H_{ik} = \{$ *1 si (i,k) est un lien sortant de i et 0 sinon*$\}$

Le diagramme (3.8) résume les différentes étapes de la Méthode de Transition Graphique (MTG). Les détails de cette méthode sont donnés en Annexe.

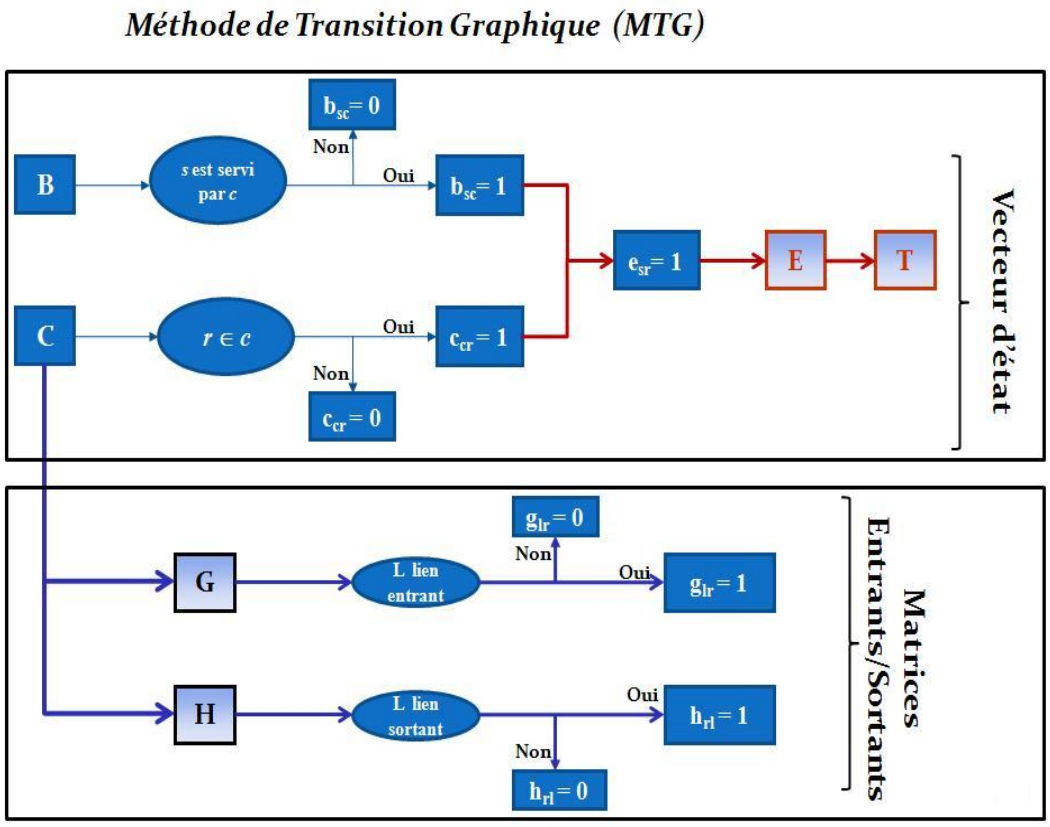


FIG. 3.8 – *Diagramme de la méthode MTG*

3.3.2.2 Suite MAC: Calcul des variables de routage

Les variables de routage représentent les fractions de paquets transmis entre les noeuds. En examinant de près l'intérieur du réseau (Fig. 3.9), on peut imaginer comment le flux de sortie du noeud est distribué vers les différents liens de sorties de ce noeud.

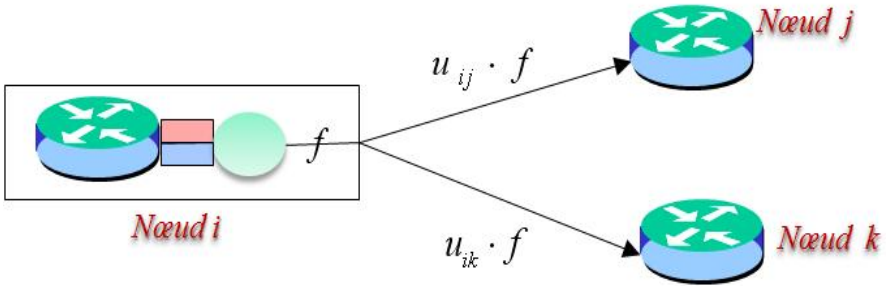


FIG. 3.9 – *Distribution des flux à l'intérieur du réseau*

Nous supposons que les chemins sont connus et pré-établis, et que la taille de l'ensemble des flux envoyés I est égale au nombre de chemins explicites utilisés par les différentes paires sources/destinations. Autrement dit, le trafic devant être acheminé entre les routeurs d'entrée et de sortie est connu à l'avance, et sa route à travers les routeurs et les liens du coeur du réseau est bien définie. Ce trafic représente le débit d'entrée du noeud source en direction du noeud destination. La distribution du trafic sur les liens de sortie dépend donc du débit d'entrée en direction du noeud destination, aussi bien que du débit de sortie de la file d'attente associée au routeur correspondant vers tous ses liens de sortie.

Soit u_{ij} la variable de routage associée au lien (i,j). Elle représente la fraction de paquets partant du noeud i vers le noeud j. *On définit la variable de routage* u_{ij} *comme étant le rapport entre la somme des flux envoyés sur le lien (i,j) et la somme des flux envoyés sur l'ensemble des liens de sortie*

du noeud i.

$$u_{ij} = \frac{\textit{Somme des flux envoyés sur le lien (i,j)}}{\textit{Somme des flux envoyés sur l'ensemble des liens de sortie du noeud } i} \tag{3.13}$$

Reprenons l'ensemble CH des chemins utilisés, et l'ensemble des liens du réseau L. Définissons la matrice A des relations "liens—chemins ", ayant en lignes l'ensemble des liens et en colonnes l'ensemble des chemins. En se basant sur l'approche graphique, et en utilisant la topologie du réseau, on calcule la matrice A comme suit:

$A = \{a_{(ij)}c, c \in CH, (i,j) \in L\}$ avec $a_{(ij)}c = 1$ *si* $(i,j) \in c$ *et 0 sinon*

Soit $I_c = I[c]$, $c \in CH$, le flux envoyé sur le chemin c, le vecteur colonne I représente les flux traversant tous les chemins utilisés par les paires sources/destinations. Son ordre est donné par le nombre de chemins utilisés. On peut ainsi calculer la somme des flux qui traverse un lien (i,j), y_{ij}, par l'équation suivante (somme des flux envoyés du noeud i en direction du noeud j):

$$A * I \Rightarrow y_{ij} = \sum_{c \in CH} a_{ij}c \; I_c \tag{3.14}$$

Reste à déterminer le deuxième élément qui intervient dans le calcul de la variable de routage, soit la somme des flux envoyés sur l'ensemble des liens de sortie du noeud i. Pour cela, on utilise la matrice H qui permet de calculer tous les liens de sortie du noeud i. La multiplication des coefficients de cette matrice par la somme des flux sortants du noeud i vers tous les autres noeuds (Eq. 3.14), permet de calculer cet élément manquant.

$$\textit{Somme des flux envoyés sur les liens de sortie du noeud } i = \sum_{k \in R, k \neq i} \sum_{c \in CH} h_{ik} \; a_{ik}c \; I_c \tag{3.15}$$

L'équation générale du calcul de la variable de routage u_{ij} est donnée par la relation suivante (d'après les 2 équations (3.14) et (3.15)):

$$\frac{A * I}{H * (A * I)} \Rightarrow u_{ij} = \frac{y_{ij}}{\sum_{k \in R, k \neq i} H_{ik}\, y_{ik}} = \frac{\sum_{c \in CH} a_{ij}c\, I_c}{\sum_{k \in R, k \neq i} \sum_{c \in CH} H_{ik}\, a_{ikc}\, I_c} \tag{3.16}$$

3.3.3 Généralisation du modèle dynamique d'état

Le travail présenté dans cette section est centré sur la conception et la réalisation d'un modèle d'état général pour les réseaux à commutation de paquets de type IP intégrant les mécanismes de différentiation de services et la commutation MPLS (IP-DiffServ-MPLS). Pour concevoir un modèle du réseau entier, les deux approches à compartiments et graphique ont été utilisés pour développer les deux méthodes MAC et MTG, qui à leur tour, ont permis de calculer les différents éléments nécessaires pour la généralisation du modèle d'état global d'un tel réseau. Les principales étapes sont les suivantes:

Etape 1: Utilisation de l'approche comportementale. Cette approche a fourni une approximation de l'évolution de l'état d'un routeur sous forme d'une équation différentielle fluide, comme étant la différence entre le flux de débit instantané d'entrée et de sortie.

Etape 2: Développement de la méthode (MAC) pour modéliser le flux de transfert entre les noeuds, en se basant sur le type de l'application visé (routage dans le réseau). Lors de cette sous-étape, on retient le modèle différentiel fluide pour estimer les flux de trafic.

Etape 3: A ce stade, la mise en ouvre de l'information sur le plan du routage devient indispensable pour représenter les interactions entre les noeuds. Par conséquent, en utilisant l'approche graphique, on conçoit la méthode MTG pour calculer ces interactions, notamment l'ordre du modèle d'état (définit par les routeurs concernés par le l'acheminement du trafic), les flux d'entrée

et de sortie d'un routeur, ainsi que les variables de routage qui réprésentent les fractions de paquets transmis sur le lien correpondant aux noeuds voisins.

Les équations (3.8),(3.10),(3.12),(3.16), permettent la généralisation de l'équation du modèle d'état pour les réseaux de type IP/MPLS. En effet, soit λ_i les débits d'entrée du routeur i, supposés connus, avec i $\in$ T: l'ensemble des routeurs concernés par l'acheminement du trafic, et ω_i ses débits de sortie vers des noeuds destinations calculés selon la méthode MAC (λ_i et ω_i peuvent être nulles), alors l'équation générale du modèle d'état est définie par l'équation (3.17).

$$\dot{x}_i = \lambda_i + \sum_{j=1,j\neq i}^{n} G_{ij}\, u_{ji}\, \mu_j\, \frac{x_j}{1+x_j} - \sum_{k=1,j\neq i}^{n} H_{ik}\, u_{ik}\, \mu_i\, \frac{x_i}{1+x_i} - \omega_i(x_i) \quad (3.17)$$

- avec H et G calculées avec la méthode MTG,

- u_{ij}: la variable de routage correspondante au lien (i,j) calculée dans la section 3.3.2.2.,

- μ_i: le taux de service de la file d'attente correpodant au routeur i supposé connu.

Ce modèle d'état offre un cadre théorique global permettant, en combinant plusieurs approches, d'offrir une modélisation générale de l'ensemble des éléments du réseau de type IP/MPLS. Ainsi, l'évolution de ces éléments, indispensable pour l'évaluation de ce genre de systèmes, est déterminée dynamiquement, et leurs valeurs sont extraites à chaque instant pour être utilisées par la commande par exemple.

Ce modèle dynamique d'état est résolu par l'intégration numérique de ces équations différentielles. A chaque pas d'intégration, les calculs suivants sont réalisés:

- Pour chaque routeur, une équation différentielle est intégrée, ou son état dynamique est recherché,

- Les informations de calcul (débit d' entrée, flux de transfert,...) sont échangées de noeud en noeud en corrélation avec les paramètres d'interconnexion (variables de routage).

3.3.4 Application sur un réseau

Considérons le réseau $G = \{R,L\}$ (Fig.3.10), de type IP/MPLS, composé de 10 noeuds (2 paires de sources/destinations et 8 noeuds du domaine MPLS). Chaque routeur i du domaine MPLS contient une file d'attente de type M/M/1/∞, avec un taux de service μ_i fixé à l'avance. Soit x_i l'état interne du routeur exprimé par sa charge en nombre de paquets. Les noeuds sont connectés entre eux par l'intermédiaire de liens unidirectionnels. Ces liens transportent les flux de transfert f_{ij} échangés entre les noeuds i et j ($i \rightarrow j$). Les mécanismes de routage explicite entre les paires sources/destinations sont représentés par des chemins qui traversent le réseau.

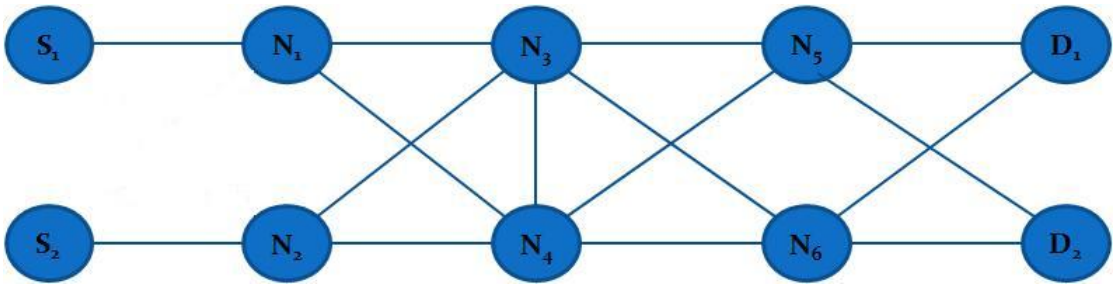


FIG. 3.10 – *Réseau G composé de noeuds et de liens*

- $R = \{S_1,S_2,N_1,N_2,N_3,N_4,N_5,N_6,D_1,D_2\}$ est l'ensemble des routeurs. Les quatre extrémités de cette topologie désignent les 2 paires sources/destinations S_1/D_1, S_2/D_2. Les noeuds N_1,....,N_6 correspondent aux routeurs MPLS,
- L est l'ensemble des liens du réseau,

- Les deux paires sources/destinations communiquent entre elles via les 4 chemins 1, 2, 3 et 4 (Fig. 3.11). Soit $CH = \{1,2,3,4\}$ l'ensemble de ces chemins possibles. Chaque chemin correspondant à un transfert de flux entre les paires sources/destinations. Soit $I = [I_1,I_2,I_3,I_4]$ le vecteur désignant les flux envoyés respectivement sur les 4 chemins possibles(1,2,3,4). Ces flux représentent les débits d'entrée du système via les deux sources S_1 et S_2.

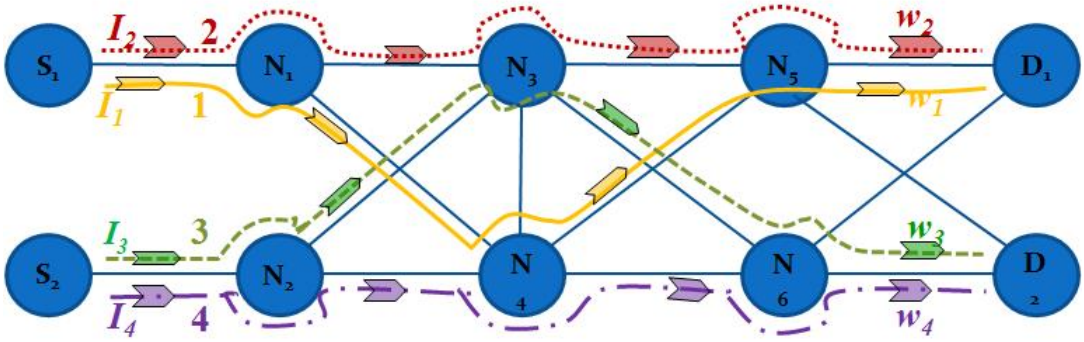


FIG. 3.11 – *Plan de routage dans le réseau G*

Calculons maintenant le système d'équations correspondant au modèle d'état de ce réseau selon la relation (3.17). Les premiers inconnus dans cette équation sont:

- l'ordre du système,
- les deux matrices G et H,
- les variables de routage,

Commençons par calculer les différentes matrices de l'algorithme de transition de la méthode MTG.

Etape 1: En utilisant l'information de routage dans le réseau, on peut calculer la matrice B de "sources/destinations—chemins".

S/D|Chemins	1	2	3	4
S_1/D_1	1	1	0	0
S_2/D_2	0	0	1	1

Etape 2: Calcul de la matrice C des relations "chemins-routeurs" basé sur le plan de routage.

Chemins|Routeurs	1	2	3	4	5	6
1	1	0	0	1	1	0
2	1	0	1	0	1	0
3	0	1	1	0	0	1
4	0	1	0	1	0	1

Etape 3: A partir de ces deux matrices, on peut calculer la matrice E des relations "sources/destinations—routeurs":

S/D|Routeurs	1	2	3	4	5	6
S_1/D_1	1	0	1	1	1	0
S_2/D_2	0	1	1	1	0	1

Cette matrice permet de déterminer l'ordre du modèle d'état. Le vecteur d'état est définit par l'ensemble T des routeurs concernés par l'acheminement du trafic: $T = \{N_1,....,N_6\}$. Le vecteur d'état est d'ordre 6 et est définit par: $x(t) = \{x_1,....,x_6\}$.

Remarque: Dans cet exemple, tous les routeurs du domaine MPLS sont concernés par le trafic, mais dans le cas d'une grande topologie, seule une partie des routeurs est concernée par l'acheminement du trafic.

Etape 4: Calcul des matrices des liens entrants G et des liens sortants H:

Matrice des liens entrants G

R|R	1	2	3	4	5	6
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	1	1	0	0	0	0
4	1	1	0	0	0	0
5	0	0	1	1	0	0
6	0	0	1	1	0	0

Matrice des liens sortants H

R|R	1	2	3	4	5	6
1	0	0	1	1	0	0
2	0	0	1	1	0	0
3	0	0	0	0	1	1
4	0	0	0	0	1	1
5	0	0	0	0	0	0
6	0	0	0	0	0	0

Calculons maintenant les variables de routage u_{ij} selon la section 3.3.2.2. Pour cela, commençons par construire la matrice A des relations "liens—chemins" en se basant sur la topologie du réseau et l'information de routage explicite qui fournit les chemins entre les paires sources/destinations.

Liens|Chemins	1	2	3	4
(1,3)	0	1	0	0
(1,4)	1	0	0	0
(2,3)	0	0	1	0
(2,4)	0	0	0	1
(3,5)	0	1	0	0
(3,6)	0	0	1	0
(4,5)	1	0	0	0
(4,6)	0	0	0	1

Ensuite, nous appliquons la relation (3.16) qui permet de calculer ces variables. Notons que les liens qui ne correspondent pas à un transfert de flux possèdent une variable de routage égale à 0. Les variables de routage sont calculées dynamiquement lors de la résolution du système d'équations. Par

exemple, les variables de routage u_{13} et u_{14}, correspondant respectivement aux liens (1,3) et (1,4), sont calculées de la façon suivante:

$$u_{13} = \frac{y_{13}}{\sum_{k=1}^{6} H_{1k}\, y_{1k}} = \frac{\sum_{c \in CH} A_{13}c\, I_c}{\sum_{k=1}^{6} \sum_{c \in CH} H_{1k}\, A_{1k}c\, I_c} = \frac{I_2}{I_1 + I_2}$$

$$u_{14} = \frac{y_{14}}{\sum_{k=1}^{6} H_{1k}\, y_{1k}} = \frac{\sum_{c \in CH} A_{14}c\, I_c}{\sum_{k=1}^{6} \sum_{c \in CH} H_{1k}\, A_{1k}c\, I_c} = \frac{I_1}{I_1 + I_2}$$

A ce stade, les paramètres nécessaires pour l'établissement de l'équation générale du modèle d'état sont établis. Le développement de l'équation (3.8) aboutit au système primaire suivant:

$$\begin{aligned}
\dot{x}_1 &= I_1 + I_2 - f_{13} - f_{14} \\
\dot{x}_2 &= I_3 + I_4 - f_{23} - f_{24} \\
\dot{x}_3 &= f_{13} + f_{23} - f_{35} - f_{36} \\
\dot{x}_4 &= f_{14} + f_{24} - f_{45} - f_{46} \\
\dot{x}_5 &= f_{35} + f_{45} - f_{5D_1} \\
\dot{x}_6 &= f_{36} + f_{46} - f_{6D_2}
\end{aligned} \tag{3.18}$$

En remplaçant les valeurs de f_{ij}, l'équation (3.17) permet d'obtenir le système d'équations suivant:

$$\begin{aligned}
\dot{x}_1 &= I_1 + I_2 - u_{13}\ \mu_1 \tfrac{x_1}{(1+x_1)} - u_{14}\ \mu_1 \tfrac{x_1}{(1+x_1)} \\
\dot{x}_2 &= I_3 + I_4 - u_{23}\ \mu_2 \tfrac{x_2}{(1+x_2)} - u_{24}\ \mu_2 \tfrac{x_2}{(1+x_2)} \\
\dot{x}_3 &= u_{13}\ \mu_1 \tfrac{x_1}{(1+x_1)} + u_{23}\ \mu_2 \tfrac{x_2}{(1+x_2)} - u_{35}\ \mu_3 \tfrac{x_3}{(1+x_3)} - u_{36}\ \mu_3 \tfrac{x_3}{(1+x_3)} \\
\dot{x}_4 &= u_{14}\ \mu_1 \tfrac{x_1}{(1+x_1)} + u_{24}\ \mu_2 \tfrac{x_2}{(1+x_2)} - u_{45}\ \mu_4 \tfrac{x_4}{(1+x_4)} - u_{46}\ \mu_4 \tfrac{x_4}{(1+x_4)} \\
\dot{x}_5 &= u_{35}\ \mu_3 \tfrac{x_3}{(1+x_3)} + u_{45}\ \mu_4 \tfrac{x_4}{(1+x_4)} - \ \mu_5 \tfrac{x_5}{(1+x_5)} \\
\dot{x}_6 &= u_{36}\ \mu_3 \tfrac{x_3}{(1+x_3)} + u_{46}\ \mu_4 \tfrac{x_4}{(1+x_4)} - \ \mu_6 \tfrac{x_6}{(1+x_6)}
\end{aligned} \tag{3.19}$$

Ce système exprime l'évolution de l'état interne de chaque routeur sous forme d'une équation différentielle fluide. Les termes de ces équations sont liées entre eux par les flux de transfert entrants ou sortants d'un noeud à un autre. La résolution de ce système se fait par intégration numérique.

Validation

La validation du modèle (3.19) se fait de la même manière que celle du modèle fluide dans le paragraphe (3.2.3.1). Le système d'équations différentielles (3.19) est résolu numériquement en utilisantet Matlab/Simulink . Il est ensuite comparé avec le modèle mathématique exact des files d'attente et avec une simulation événementielle (Language Tcl). Les paramètres de simulation sont les suivants:

- Le temps de simulation est de 100 secondes,

- Les taux de service des files d'attente correspondant à chaque routeur sont fixés

$\mu_1 = 7$, $\mu_2 = 9$, $\mu_3 = 8$, $\mu_4 = 10$, $\mu_5 = 8$, $\mu_6 = 10$

- Les valeurs des débits d'entrées du système sur les différents chemins sont données par:

$I_1 = 4$, $I_2 = 2$, $I_3 = 5$, $I_4 = 3$

Les taux de service, débits d'entrée et de sortie, ainsi que les flux de transfert f_{ij} sont estimées en paquets/période de temps.

Le tableau 3.12 présente les résultats de la comparaison des 3 méthodes. On remarque que les valeurs obtenues par la résolution analytique du modèle fluide sont, dans la plupart des cas, identiques à celles obtenues par simulation du modèle de la théorie des files d'attente (colonne 2), et plus précises que celles obtenues avec une simulation événementielle. On conclut ainsi que le modèle fluide aboutit à une approximation très précise de l'évolution de l'état de charge des routeurs du réseau modélisé.

	Résolution analytique du modèle fluide	Calcul selon la théorie de files d'attente	Simulation événementielle
$x_1(t)$	6.000	6.000	6.066
$x_2(t)$	8.000	8.000	8,143
$x_3(t)$	7.000	6,999	7,124
$x_4(t)$	2,333	2,333	2,353
$x_5(t)$	3,000	3,000	2,975
$x_6(t)$	4,000	4,000	3,999

FIG. 3.12 – *Comparaison des résultats obtenus avec les 3 méthodes*

Pour plus d'expérimentations nous avons inséré des perturbations déterministes qui peuvent être modélisées par des entrées instables, en particulier avec des entrées aléatoires. En effet, si le modèle est construit pour la synthèse des applications de loi de commande ou pour la gestion de contrôle, la prise en considération de l'existence d'une perturbation pendant la phase de modélisation peut améliorer les performances de l'application, en particulier la stabilité du système face à ces perturbations. Nous avons simulé le système (3.19) avec des entrées variant en fonction du temps. Nous avons choisi des entrées aléatoires afin de tester la réponse du système (3.19) à un changement de trafic brusque lors de l'étape de simulation. Nous avons remarqué que le système reste stable face à ces changements, en assurant toujours une capacité en temps réel à échanger les flux de transfert entre les composants du réseau correspondant. Les figures 3.13 et 3.14 montrent respectivement l'entrée aléatoire et le comportement de l'évolution de l'état de charge $x_1(t)$ effectué par une résolution analytique du modèle fluide. La figure 3.14 montre que le système reste stable face à ses changements, en assurant toujours une capacité en temps réel à échanger les flux de transfert entre les composants du réseau correspondant. On remarque aussi que le comportement de l'état

de charge x_1(t) est en corrélation permanente avec les valeurs de l'entrée, et il suit le même rythme du trafic.

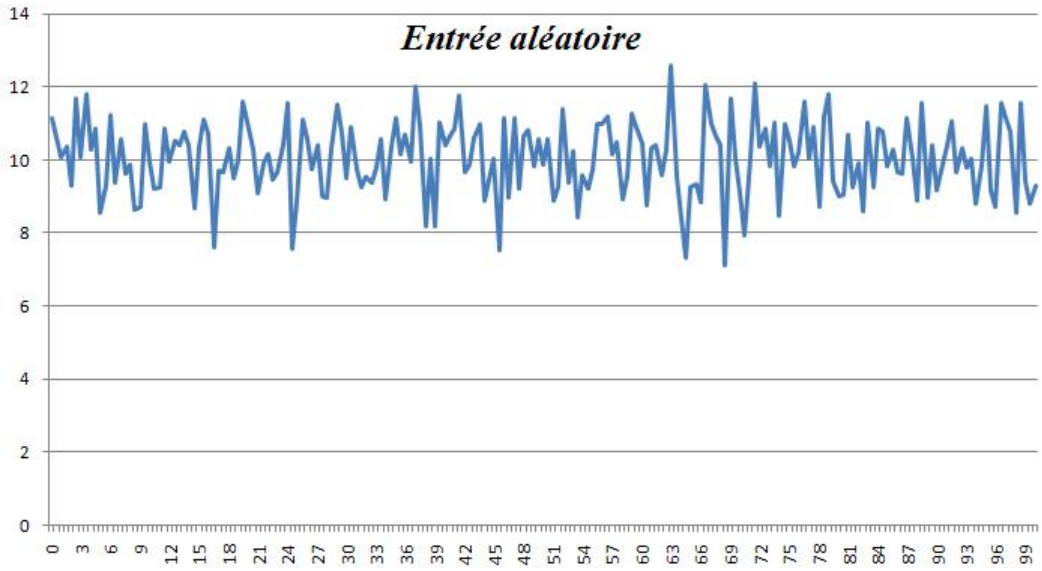


FIG. 3.13 – *Entrée aléatoire*

3.4 Modèle d'état du réseau

Dans la section précédente, nous avons développé un modèle d'état dynamique, décrivant le comportement des principaux éléments du réseau de type IP/MPLS. Nous l'avons ensuite résolu analytiquement en fonction de l'état de charge de ses routeurs, des débits d'entrée et des capacités de traitement des routeurs.

Dans cette section, notre objectif consiste à mettre le système d'équations développé sous forme de représentation d'état.

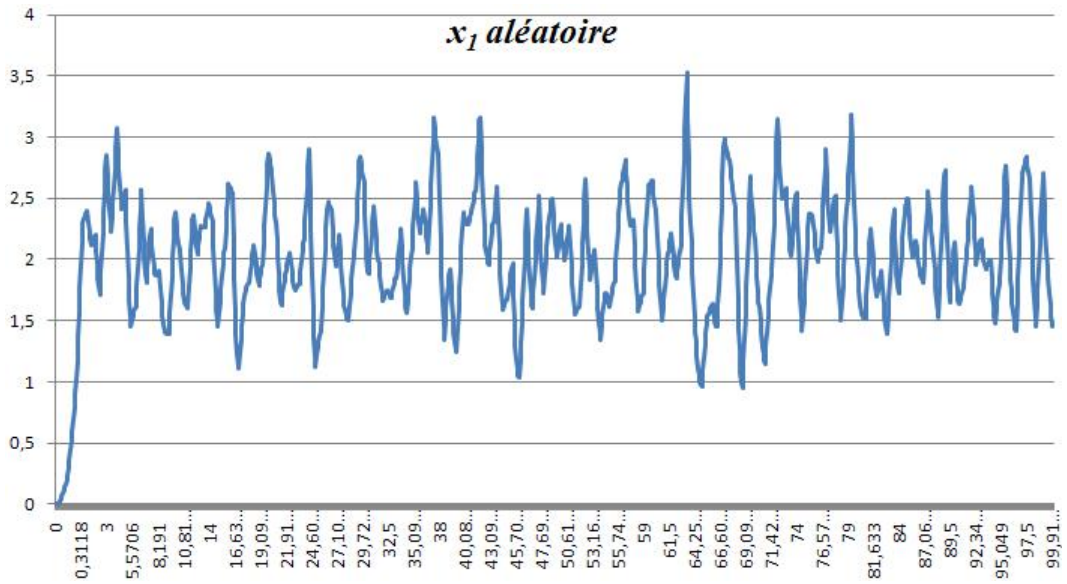


FIG. 3.14 – *Evolution de l'état de charge $x_1(t)$ pour une entrée aléatoire*

3.4.1 Systène non linéaire sans retard

Le système d'équations (3.17) peut être représenté par le système non linéaire (3.20), avec $y_i(t) \cong w_i(x_i(t))$ qui représentent les flux de sortie:

$$\begin{aligned} \dot{x}(t) &= f(x,u,t) \\ y(t) &= h(x,t) \end{aligned} \tag{3.20}$$

avec f, h deux fonctions continues dans $\mathbb{R}_+$

Considérons le modèle possédant les dimensions suivantes:

- L'ensemble T contient n routeurs concernés par le trafic : $[T] = n$,
- L'ensemble I contient k entrées, correspondantes aux sources,
- L'ensemble W contient w sorties correspondentes aux destinations.

Le système (3.20) peut se mettre sous forme d'un système non linéaire de la forme suivante:

$$\begin{aligned} \dot{x}(t) &= A(x)x(t) + Bu(t) \\ y(t) &= C(x)x(t) \end{aligned} \tag{3.21}$$

Avec:

- $x = (x_i,\ i = 1,...,n)$: un vecteur colonne correspondant aux états de charge des routeurs de l'ensemble T,

- $u = (I_i,\ i = 1,...,k)$ un vecteur colonne correpsondant aux entrées du système,

- $A = [n * n]$: matrice d'état représentant l'état de charge du routeur,

- $B = [n * k]$: matrice de commande ou d'entrée, elle traduit les connections entre les routeurs et les entrées du système,

- $C = [w * n]$: matrice de sortie, elle représente les connections entre les routeurs et les sorties.

3.4.2 Système non linéaire avec retard

Le retard est fréquemment rencontré dans les systèmes technologiques et peut affecter leur fonctionnement de manière significative. Dans un système dynamique sans retard, la seule connaissance de l'état x à l'instant t_0 permet de connaître l'état à tout instant t. Par contre si on tient compte du retard sous la forme suivante:

$$\dot{x}_i = x_i(t - \tau) \qquad \text{i=1.....6}$$

Alors la simple connaissance de $x(t_0)$ n'est plus suffisante pour connaître l'évolution du système, pour cela il faut connaître la valeur de l'état à chaque instant de l'intervalle [t_0-τ, t_0].

Les retards apparaissent naturellement dans les modèles des processus réels. Dans les réseaux des communications, les principales sources sont les phénomènes de stockage et de transport de flux de trafic (traitement des files d'attente, transmission des flux..). Cependant, les retards peuvent être aussi introduits artificiellement. Dans ce travail, on restreint l'étude des retards au

niveau des files d'attente des routeurs du réseau en fonction de leurs vitesses de traitement, en supposant que les retards dus aux délais de transmission sont négligeables. En effet, chaque routeur contient une file d'attente qui sert d'espace de stockage des flux arrivants, avant de les traiter et les livrer aux routeurs voisins. Cette capacité de traitement peut être influencée par différents facteurs extérieurs ou intérieurs au routeur. Parmi eux on trouve le taux de service de la file d'attente. Nous allons donc considérer un retard distribué au niveau de la vitesse de traitement de la file d'attente pour chaque routeur du réseau.

Le système (3.21) aura la forme suivante :

$$\begin{aligned} \dot{x}(t) &= A(x)x(t-\tau) + Bu \\ y(t) &= C(x)x(t-\tau) \end{aligned} \tag{3.22}$$

Avec $x(t_0 - \tau) = x(t_0)$, et les mêmes définitions de A, B et C que (3.21).

Le retard au niveau des taux de traitement des files d'attente entraîne une accumulation des flux dans les routeurs d'entrée, et ainsi un taux de livraison plus faible vers les routeurs voisins que dans le cas normal sans retard. Les conséquences de ce retard sur l'état de charge des routeurs sont exprimées par le rapport entre les taux de transfert des flux et la capacité de traitement des files d'attente. En effet, l'état de charge des routeurs peut augmenter ou diminuer selon ce rapport. Cependant, la théorie fondamentale des files d'attente [40] montre que la durée moyenne de traitement est équivalente aux taux de service μ:

$$\textit{La durée moyenne de service} = 1/\mu$$

D'où, afin de mieux gérer le retard variable, on utilise la fonction *"variable transport delay"* du logiciel *Matlab* [67], qui permet d'introduire un retard sur chaque état x_i des routeurs du réseau, correspondant à la durée moyenne de service de sa file d'attente:$1/\mu_i$.

Dans la suite, nous allons comparer le modèle fluide global avec et sans retard, tout en utilisant les mêmes paramètres d'entrées, de taux de service que dans le paragraphe 3.3.4. La figure 3.15 montre l'évolution de l'état de charge du routeur N_1 dans les deux cas. Vu que N_1 est un routeur d'entrée du réseau (routeur dans lequel le trafic entre dans le réseau), on remarque que l'accumulation de flux dans le cas sans retard est plus faible que dans le cas avec retard. Cette accumulation est due à la faible capacité de traitement de la file d'attente correspondante au routeur N_1. Par contre, comme illustré par la figure 3.16, correspondante à l'évolution de l'état de charge du noeud N_3, le taux d'accumulation de flux dans le cas avec retard est plus faible que celui dans le cas sans retard. En effet, étant donné que N_3 est un noeud du coeur du réseau, le taux de livraison du flux est affaiblit à chaque traversée d'un routeur du chemin de l'entrée vers N_3. Cette conclusion pousse à croire que le retard influence l'évolution de l'état de charge du réseau. On remarque également que le retard n'est pas uniforme sur tous les noeuds du réseau.

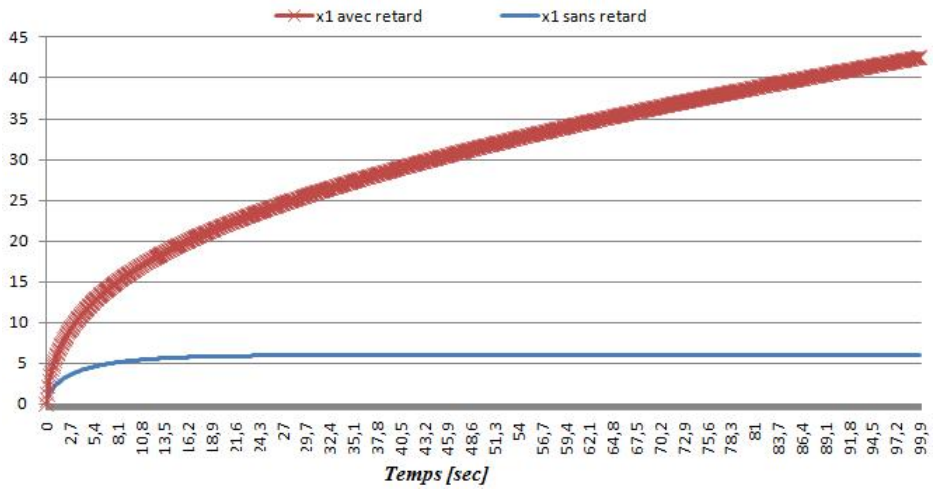


FIG. 3.15 – *Evolution de l'état de charge $x_1(t)$ avec et sans retard*

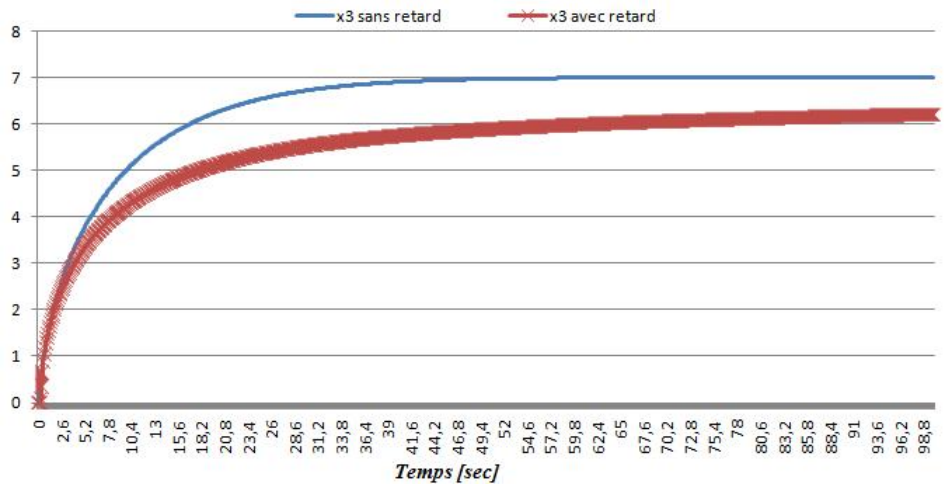


FIG. 3.16 – *Evolution de l'état de charge $x_3(t)$ avec et sans retard*

3.5 Conclusion

Nous avons présenté un modèle dynamique d'état d'un réseau de type IP/MPLS. Nous avons considéré que le modèle fluide est le plus approprié pour étudier le comportement d'un tel réseau. Ce modèle fluide rend possible la diminution de la complexité et du coût de son implémentation, tout en offrant une capacité d'approximation de l'état de charge du réseau et de mesurer la bande passante disponible sur ses chemins de bout en bout.

Nous avons commencé par modéliser le trafic qui traverse un noeud du réseau à l'aide d'une approche mathématique appelée approche à compartiments. Cette approche a permis de proposer l'équation du trafic qui exprime l'évolution de l'état de charge du noeud par une équation différentielle fluide.

La représentation des caractères topologiques ainsi que la connectivité du réseau est devenue indispensable pour la mise en oeuvre du modèle fluide global. Pour cela, en se basant sur une approche graphique, nous avons développé les deux méthodes, Méthodes d'Assemblage à Compartiments (MAC) et Méthode de Transition Graphique (MTG), qui permettent de modéliser

les interactions entre les différents composants du réseau et les insérer dans l'équation correspondante à chaque routeur du réseau.

L'ensemble des équations de tous les routeurs ainsi obtenues forment un système d'équations dynamiques, qui résume l'évolution de tous les éléments du vecteur d'état. Ce système a été validé par des simulations de la théorie des files d'attente et par des simulations événementielles. Les résultats ont montré une très bonne approximation du système fluide.

Enfin, nous avons montré que le retard influence l'évolution du système.

Le modèle proposé dans ce chapitre présente plusieurs intérêts, comme celui de la représentativité. En effet, ce modèle est capable de représenter les interconnexions entre les différents composants du réseau d'une manière générale. L'autre point fort du modèle correspond à la dynamicité de ses équations. En effet, l'évolution dynamique de ses paramètres, qui soient les éléments du vecteur d'état correspondant ou les variables de routage, permet de connaître à chaque instant l'état de charge du réseau ainsi que de mettre à jour le plan de routage établi. Ce modèle servira de base pour étudier le problème de congestion du réseau.

Chapitre 4

Systèmes de contrôle dans un réseau IP/MPLS

L'objectif de ce chapitre est de proposer des algorithmes de commande d'un réseau de type IP/MPLS permettant de bien gérer les ressources en respectant au mieux les besoins des demandes de trafic et la bande passante requise tout en évitant les phénomènes de congestion. Pour cela quelques questions se posent:

- Comment introduire un mécanisme de contrôle dans un réseau IP/MPLS, déjà connu par son choix sélectif des chemins explicites pouvant transporter le trafic du réseau?
- Quelles sont les métriques à adopter dans le processus de contrôle?
- Comment obtenir des informations sur les ressources disponibles dans le réseau?

4.1 Contexte général

Les réseaux IP/MPLS offrent une meilleure gestion des ressources tout en garantissant une bonne qualité de service. Cependant, l'expansion grandissante d'Internet a entraîné une saturation des réseaux. La congestion peut être provoquée dans plusieurs cas parmi lesquels, celui d'un émetteur qui envoie des données à des vitesses trop élevées pour le récepteur et qui dépassent sa capacité de gestion, le cas où les ressources réseaux sont inférieures à la demande, ou encore le cas d'un déséquilibre du trafic dans un domaine du réseau. Cependant, une solution possible doit tenir compte de la capcité du réseau (capacité des routeurs internes, bandes passantes des liens..). D'où la nécessité d'adapter les méthodes de contrôle au type du réseau étudié.

De façon générale, une transmission de données sur une ligne de communication dépend de l'évolution de l'état de charge du réseau, surtout dans le cas des réseaux de type IP/MPLS. Au fait, ces réseaux s'appuient sur les informations des ressources disponibles (Bande passante, délai, files d'attente..), afin d'établir la connexion entre les différents couples récepteurs/émetteurs. Cependant, un mécanisme de contrôle doit donc tenir compte des métriques internes au système, afin de garantir au mieux les performances des réseaux, notamment en terme de respect de la bande passante. Un tel mécanisme serait capable d'estimer l'état du réseau et de réagir face à une congestion, tout en suivant l'évolution dynamique des données présentes sur une ligne de transmission.

Il existe plusieurs travaux traitant la gestion de contrôle de congestion dans les réseaux. Certains se basent sur les connexions bouts-en-bouts (i.e., les informations du contrôle sont échangées seulement entre l'émetteur et le récepteur) ou sur les connexions points-à-points (i.e., échange des informations entre les composants internes du réseau notamment les routeurs).

Par ailleurs, l'idée est de proposer un modèle complet qui tient compte de la dynamique totale d'une ligne de communication depuis l'émetteur jusqu'au récepteur, en passant par les différents composants internes du réseau notamment les liens et les routeurs. En effet, un tel modèle considère non plus une seule connexion mais une ligne entière émetteur-routeurs-liens-récepteur.

Dans le chapitre 3, nous avons détaillé une proposition d'un modèle analytique global pour les réseaux IP/MPLS, à base duquel toutes les dynamiques peuvent être observées et calculées à tous les niveaux d'une communication (l'entrée des routeurs, charge des files d'attente, flux transmis par les routeurs, etc.), et qui nous servira pour élaborer des mécanismes de contrôle.

4.2 Contrôle dans les réseaux IP/MPLS

L'étude réalisée dans le chapitre 3, nous a permis de choisir la méthode de prédiction des ressources qui semble la plus efficace, celle basée sur la modélisation analytique. En effet, il nous semble plus simple et plus efficace de manipuler les contraintes de QdS requises par les applications et déduire les performances globales atteintes par le réseau, en se basant sur un modèle analytique. Par contre, même si l'étude des performances du réseau dans les conditions de saturation, donne les frontières et les limites asymptotiques fondamentales du débit du système, elle ne peut pas donner les meilleures conditions de fonctionnement du réseau et par suite les valeurs optimales de performance. Cependant, il est nécessaire de bâtir un mécanisme de contrôle capable de gérer son fonctionnement.

4.2.1 Modèle analytique de base

Dans le chapitre précédent, nous avons développé un modèle analytique capable de prédire les métriques de performance des réseaux IP/MPLS. Les travaux menés dans le cadre de ce chapitre visent, en utilisant le modèle analytique, à introduire de nouveaux mécanismes de contrôle afin de mieux gérer les ressources des réseaux et éviter ainsi une situation de saturation ou de congestion.

Dans cette section, nous considérons les mêmes caractéristiques du réseau décrit dans la section 3.3.1. Ce réseau a été modélisé dans le chapitre 3 par l'équation générale suivante:

$$\dot{x}_i = \lambda_i + \sum_{j=1, j\neq i}^{n} G_{ij}\, u_{ji}\, \mu_j\, \frac{x_j}{1+x_j} - \sum_{k=1, k\neq i}^{n} H_{ik}\, u_{ik}\, \mu_i\, \frac{x_i}{1+x_i} - \omega_i(x_i) \quad (4.1)$$

4.2.2 Choix et calcul des métriques de contrôle

Le réseau IP/MPLS est caractérisé par de nombreuses applications, dont celle du routage explicite. Le principe du routage explicite consiste à choisir des chemins établis à l'avance entre une paire source/destination, en se basant sur les ressources du réseau. Dès lors, surveiller ces ressources devient une tâche primordiale pour le bon fonctionnement du système, afin d'agir et empêcher toute sorte de congestion ou de saturation. Cependant, les hypothèses des conditions de saturation sont nécessaires pour l'analyse et la modélisation du réseau. Prenons l'hypothèse que les files d'attente contiennent toujours suffisamment de données à transmettre sur chaque lien du réseau, avec une grande capacité de stockage. Cette hypothèse permet de négliger l'étude des dynamiques de la file d'attente au profit d'autres métriques plus importantes dans le cas de l'étude des réseaux IP/MPLS, même s'il reste tou-

jours utile pour le calcul de ses métriques. Elle nous éloigne aussi du besoin de la modélisation détaillée de la nature du trafic, de ses caractéristiques et de son taux d'arrivée au niveau des files d'attente.

Par ailleurs, en rassemblant ces hypothèses, la métrique la plus importante à prendre en compte dans le cas de l'étude des réseaux IP/MPLS, est sans doute la bande passante des chemins de transfert du trafic entre les paires sources/destinations, composés en outre par les bandes passantes des liens du réseau. La considération de cette métrique est essentielle car la saturation des bandes passantes provoque la congestion du réseau, et ainsi la perte des données, en plus elle constitue un paramètre de base de la gestion de différents protocoles basés sur les réseaux IP/MPLS.

La bande passante d'un chemin, notée BW_c avec $c \in CH$ (ensemble de chemins défini dans le chapitre 3), représente la capacité de ses liens à transférer le trafic du réseau. Donc, l'évolution de cette bande passante permet de calculer le reste de la capacité du chemin à transporter des données lors de la prochaine demande, c'est ce qu'on appelle la bande passante résiduelle du chemin "BWR_c" , $c \in CH$. Pour calculer cette bande passante résiduelle, nous utilisons le modèle analytique développé dans le chapitre précédent. Ce modèle nous a permis de calculer l'évolution du flux de transfert du trafic "f_{ij}" sur le lien entre deux noeuds i et j du réseau (i,$j \in R$ ensemble des routeurs défini dans le chapitre 3). A partir de la valeur de f_{ij}, on peut calculer à tout instant, la bande passante résiduelle du lien (i,j) correspondant, notée BWR_{ij}, en faisant la différence entre la bande passante initiale du lien, notée BW_{ij}, et la valeur de f_{ij}. Ainsi, la bande passante résiduelle du chemin BWR_c est calculée comme étant le minimum de bandes passantes résiduelles des liens de ce chemin. De cette façon, on est capable de calculer exactement, à chaque instant, la bande passante résiduelle de tous les liens et déduire la

bande passante du chemin composé de ces liens en appliquant les relations suivantes:

$$\begin{gathered} Pour (i,j) \in L : ensemble\ de\ liens \quad et\ c \in CH : ensemble\ de\ chemins \\ BWR_{ij} = BW_{ij} - f_{ij} \\ BWR_c = Min\{BWR_{ij}, \quad (i,j) \in c\} \end{gathered} \tag{4.2}$$

4.2.3 Modèle de contrôle de congestion basé sur l'information de la bande passante résiduelle

Dans cette partie, nous étudions le problème de congestion dans le cas où la demande du trafic au niveau de l'entrée du réseau dépasse la capacité de transfert de ce dernier. Cette capacité est exprimée par les bandes passantes des chemins du réseau. En effet, la capacité résiduelle de ces chemins représente un indicateur de congestion du réseau, dans le cas où les flux de trafic envoyés depuis les sources doivent traverser ces chemins pour être livrés aux destinations. Donc, l'objectif est de surveiller et contrôler la bande passante résiduelle de ces chemins afin d'éviter la congestion et la saturation du réseau.

Dans cette section, nous nous inspirons d'une étude faite sur le contrôle de congestion dans les réseaux à compartiments [7]. Cette étude s'intéresse au contrôle de congestion au niveau des files d'attente des routeurs. Notre objectif est de transformer ce modèle de contrôle pour qu'il soit adapté aux chemins d'un réseau IP/MPLS.

Nous considérons les mêmes caractéristiques d'un réseau à compartiments comme défini dans la section 3.3.1. En effet, supposons que le réseau comporte n compartiments, m chemins et l'équivalent de m en flux d'entrée. Ce

réseau possède les mêmes définitions et propriétés que dans la section 3.3.1. Nous supposons qu'il est CCE et CCS comme définit dans la section 3.3.1 (chaque compartiment est connecté à au moins une entrée et une sortie). Cette propriété permet d'installer des entrées (sources) et des sorties (destinations) au réseau, ainsi que d'établir un certain nombre de chemins entre les paires sources/destinations. Les liens du réseau sont caractérisés par une capacité maximale de transfert des flux f_{max}, équivalente à leur bande passante initiale. Ainsi, la capacité de transfert maximale et résiduelle (BWR) d'un chemin du réseau est calculée selon la méthode (4.2). Nous supposons aussi que les compartiments du réseau possèdent une capacité de stockage supérieure à celle des liens du réseau. La demande du trafic initiale, notée par d_i, est injectée par la source i sur un chemin du réseau (le chemin est choisit par hypothèse). En effet, cette demande sera contrôlée en fonction de la bande passante résiduelle du chemin, où d_i sera assignée au débit d'entrée réel envoyé sur le chemin et noté par I_i. C'est le même principe que celui considéré dans [7], à l'exception de la demande du trafic initiale qui sera contrôlée en fonction de la bande passante résiduelle des chemins et non pas selon la capacité des routeurs du réseau.

Dans le chapitre 3, l'analyse de ce réseau a permis de construire un modèle dynamique d'état selon l'équation différentielle (4.1). Ce modèle d'état s'écrit de la manière suivante sous forme matricielle:

$$\begin{aligned} \dot{x} &= A(x)x + Bu \\ y &= C(x)x \end{aligned} \tag{4.3}$$

Avec:

- x est le vecteur d'état, u est le vecteur d'entrée et y le vecteur de sortie,
- A, B et C représentent respectivement les matrices d'état, d'entrée et de sortie.

(ces matrices sont calculées selon la section 3.4.1)
Ce modèle permet de calculer, à chaque instant, et d'une manière dynamique, les flux de transfert f_{ij} entre deux noeuds i et j. A partir de ces valeurs, et en connaissant les bandes passantes initiales des liens, on sera capable de calculer à chaque instant les bandes passantes résiduelles des liens et en conclure celles des chemins du réseau selon la méthode (4.2). Dans notre cas, les mesures disponibles pour le contrôle en rétroaction sont les valeurs des bandes passantes résiduelles des chemins BWR_c, $c \in CH$, calculées à partir de l'évolution des flux de transfert sur les liens du réseau. Si les bandes passantes résiduelles diminuent, la capacité du réseau à transférer des données diminue aussi, et on se rapproche de plus en plus d'une situation de saturation des liens, ce qui provoque une congestion du réseau. Pour résoudre ce problème, un système de contrôle en rétroaction est introduit sur les chemins utilisés pour l'acheminement du trafic dans le réseau, afin de contrôler les flux de données envoyés sur ces chemins et d'éviter ainsi que leurs bandes passantes résiduelles ne se dégradent en provoquant la congestion du réseau.

Comme dans [7], le mécanisme de contrôle fonctionne de la manière suivante: les demandes initiales (d_i) sur les chemins du réseau sont assignés à des valeurs inférieures (I_i), selon les valeurs des bandes passantes résiduelles de ces chemins (Equation 4.4):

$$I_i(t) = \beta_i(t) * d_i(t) \qquad 0 \leq \beta_i(t) \leq 1 \tag{4.4}$$

Où $\beta_i(t)$ représente la fraction de la demande initiale $d_i(t)$ envoyée sur un chemin du réseau.

Les valeurs de β_i seront proportionnelles à celles des bandes passantes résiduelles des chemins et tendent vers zéro en cas de saturation des liens.

Par ailleurs, le contrôle proposé dans [7] est modifié pour intégrer les

bandes passantes résiduelles des chemins du réseau de la façon suivante:

$$\begin{aligned} \dot{z}_i &= y_i - \phi(z_i)\,\alpha_i\,d_i \qquad i \in I_{out} \\ \beta_i(z) &= \alpha_i\,\phi(z_i) \qquad i \in I_{in} \end{aligned} \tag{4.5}$$

Avec:

1- I_{in} est l'index des entrées ($|I_{in}| = m$)

2- I_{out} est l'index des chemins ($|I_{out}| = m$)

3- α_i, $i \in I_{out}$ sont des paramètres de synthèse tels que: $0 \leq \alpha_i \leq 1$

4- ϕ: $\mathbb{R}_+ \rightarrow \mathbb{R}_+$ est une fonction monotone croissante telle que $\phi(0) = 0$ et $\phi(+\infty) = 1$,

5- Le paramètre y_i correspond à la bande passante résiduelle du chemin i, obtenu par la méthode (4.2).

L'établissement du modèle de contrôle est illustré dans [7] par la figure (4.1).

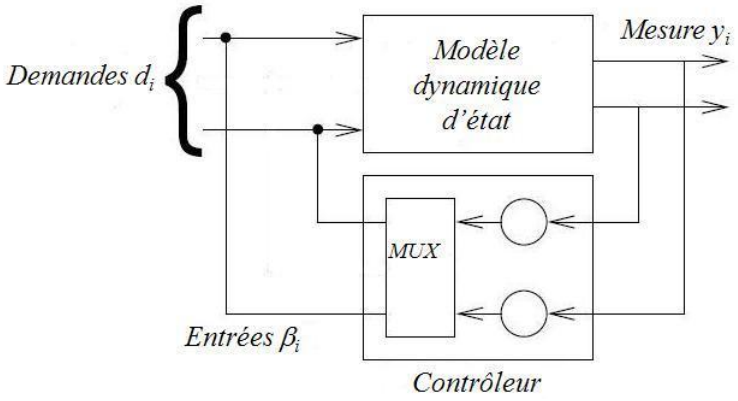


FIG. 4.1 – *Structure du modèle de contrôle*

Dans [7], les auteurs représentent le contrôleur par une structure de réseau à compartiments. Dans notre cas, son ordre sera défini avec autant de compartiments qu'il y a de chemins dans le réseau à contrôler. Ainsi, chaque entrée du contrôleur est liée à l'une des bandes passantes résiduelles des che-

mins, et chaque sortie du contrôleur définit les paramètres de contrôle β_i (Fig.4.1).

En effet, dans [7] les auteurs précisent que les contrôles $\beta_i(z)$ (les fractions de la demande d'entrée qui sont calculées par le contrôle) sont bien confinés dans l'intervalle [0, 1]. Sous la condition (4) ci-dessus, ils expriment que si 0 $\leq \phi(z_i) \leq 1 \; \forall \; z_i \in \mathbb{R}_+$, et avec la condition (3), on aura:

$$0 \leq \beta_i(z) = \alpha_i * \phi(z_i) \leq \alpha_i \leq 1$$

La congestion du réseau est alors évitée. En effet, le contrôle assure que les débits envoyés sur les chemins soient inférieurs à la capacité de tansfert maximale de ces chemins.

4.2.3.1 Exemple

La topologie du réseau utilisée pour valider le modèle de contrôle est la même que celle décrite dans le paragraphe 3.3.4 (Fig. 4.2).

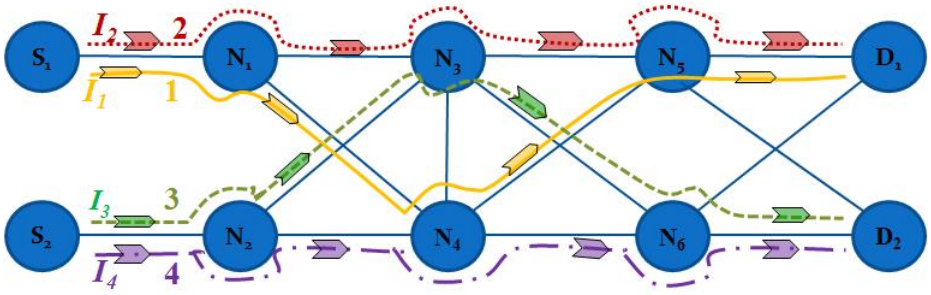


FIG. 4.2 – *Représentation des caractéristiques du réseau*

Ce réseau possède les mêmes définitions et propriétés que celles définies dans la section 3.3.1. Un système d'état dynamique a été établi dans le section 3.3, il décrit l'évolution de ce système et analyse ses paramètres essentielles comme le taux de transfert des flux, la taille des files d'attente, etc.

Dans ce paragraphe, nous présentons un exemple numérique du modèle de contrôle (4.5), qui sera lié au système d'équations principal par les variables de contrôle ainsi calculées. D'après le système (4.5), le modèle dynamique correspondant au contrôleur a la forme suivante:

$$\begin{array}{lcll} \dot{z}_1 & = & y_1 - \phi(z_1)\,\alpha_1\,d_1 & \beta_1(z) = \alpha_1 * \phi(z_1) \\ \dot{z}_2 & = & y_2 - \phi(z_2)\,\alpha_2\,d_2 & \beta_2(z) = \alpha_2 * \phi(z_2) \\ \dot{z}_3 & = & y_3 - \phi(z_3)\,\alpha_3\,d_3 & \beta_3(z) = \alpha_3 * \phi(z_3) \\ \dot{z}_4 & = & y_4 - \phi(z_4)\,\alpha_4\,d_4 & \beta_4(z) = \alpha_4 * \phi(z_4) \end{array} \tag{4.6}$$

Conformément aux règles décrites dans la section précédente, nous choisissons les paramètres suivants :

- Les paramètres μ correspondant aux taux de service des files d'attentes sont fixés à 30,

- Les valeurs des demandes sont générées aléatoirement,

- Les valeurs des paramètres de synthèse α_i associés à chaque chemin sont supposées égales aux valeurs des variables de routage correspondant à ces chemins. Mais contrairement au [7], leurs valeurs seront calculées comme dans le paragraphe 3.3.2.2, c.à.d.:

$$\alpha_1{=}u_{14},\ \alpha_2{=}\ u_{13},\ \alpha_3{=}\ u_{23},\ \alpha_4{=}u_{24}$$

- Comme dans [7], la fonction ϕ est définie par : $\phi(z){=}\ \frac{z}{\epsilon+z}$, avec $\epsilon = 10^{-3}$, afin de satisfaire la condition (4) du modèle (4.5),

- Le paramètre y_i correspond à la bande passante résiduelle du chemin i, et calculé selon la méthode (4.2).

La figure (4.3) illustre la structure générale du réseau contrôlé:

Dans les graphes (4.4) à (4.7), nous considérons l'évolution et le contrôle du chemin 2. Rappelons que le principe est le même pour les autres chemins. Les demandes du trafic initial sur le chemin 2 sont représentées par

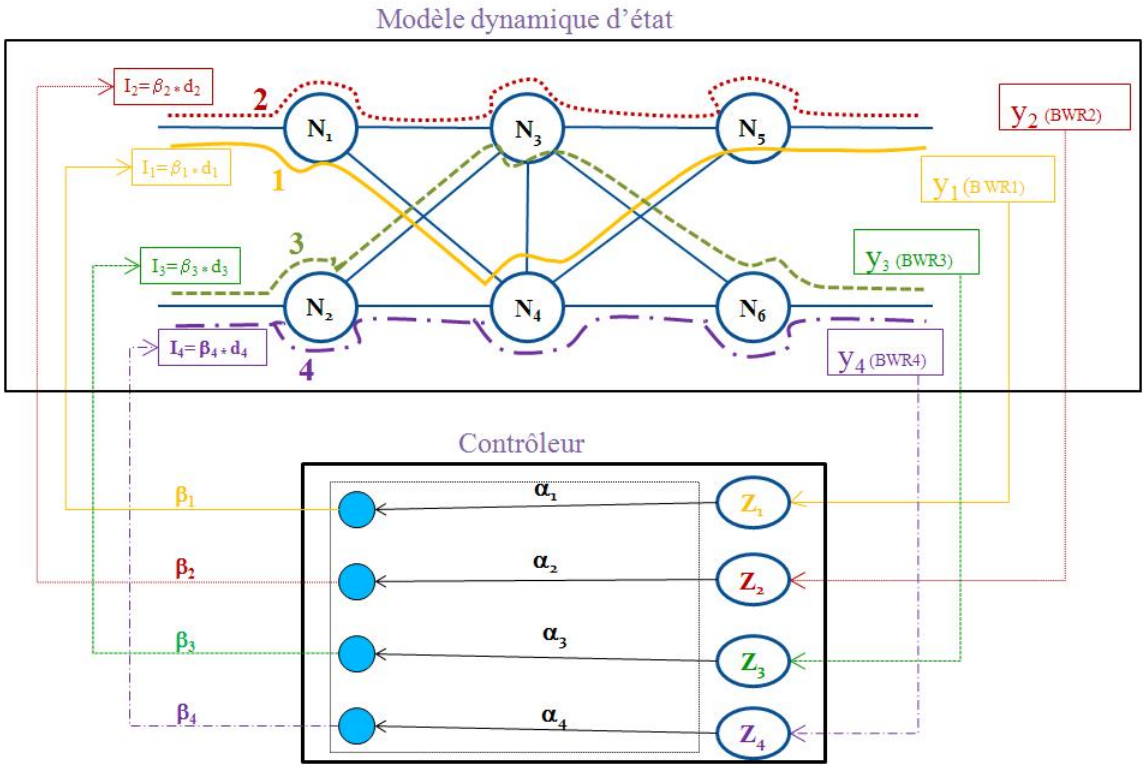


FIG. 4.3 – *Structure générale du réseau contrôlé*

d_2 (Fig.4.4). C'est un trafic aléatoire qui change en fonction du temps. Cependant, on observe sur la Fig.4.5, que la bande passante résiduelle varie en fonction de l'évolution de la demande. Donc afin d'adapter les demandes avec le reste de la bande passante de ce chemin, le contrôle agit en fractionnant les demandes du trafic (Fig. 4.7) par les valeurs de la variable de contrôle β_2 (Fig. 4.6). En effet, on remarque que les valeurs de β_2 (Fig. 4.6) sont toujours comprises entre 0 et 1, et qu'elles varient proportionnellement par rapport à l'évolution de la bande passante résiduelle, c.à.d. que β_2 decroît vers 0 quand la bande passante résiduelle decroît afin d'affiner la demande correspondante, et à l'inverse β_2 croît quand la bande passante résiduelle croît afin de laisser circuler plus de trafic.

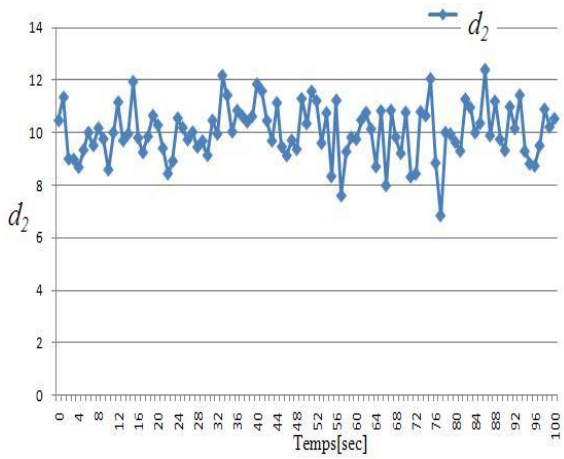


FIG. 4.4 – *Demande de trafic* d_2

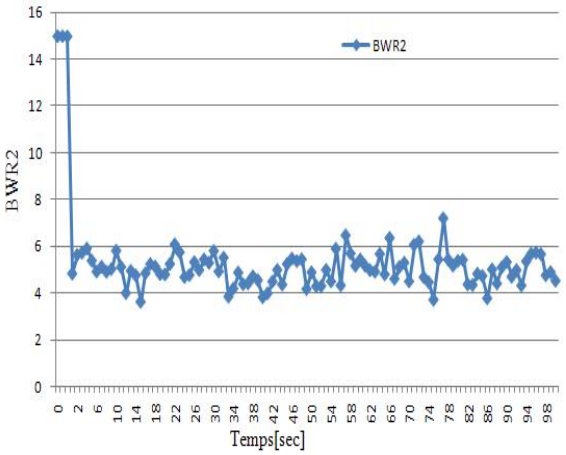


FIG. 4.5 – *Bande passante résiduelle*

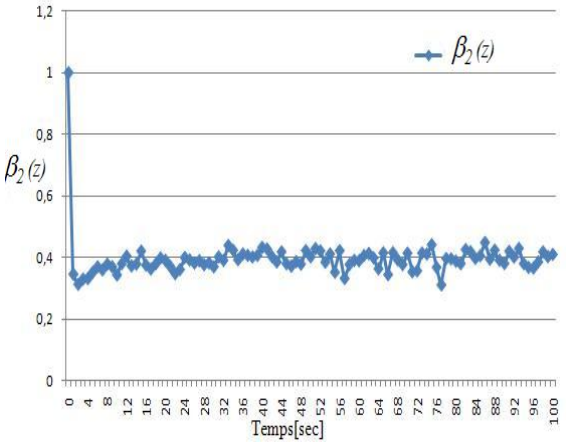


FIG. 4.6 – *Paramètre de contrôle* β_2

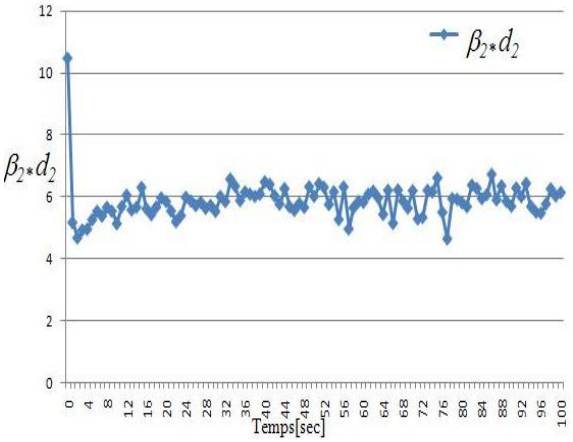


FIG. 4.7 – *Trafic contrôlé*

4.2.4 Modèle de contrôle par routage multi-chemins

Dans les systèmes de communications, les protocoles cherchent à créer, pour chaque couple de noeuds souhaitant communiquer, un chemin ou une route les reliant. Certains de ces protocoles étendent les mécanismes connus afin de définir non pas un seul, mais plusieurs chemins entre chaque couple. On parle alors de routage multi-chemins. Rappelons que deux modèles de routage multi-chemins LBWDP et PEMS ont été étudiés dans les deux premiers chapitres. Ces modèles à description multiple fournissent une nouvelle approche de représentation. Ils permettent également de lutter contre les phénomènes de congestion. A présent, on sait que les réseaux de type IP/MPLS offrent une grande capacité à garantir l'existence de plusieurs routes entre les couples de noeuds communicants. Cependant, une route entre deux noeuds peut dès lors disparaître à tout moment (surcharge des liens, panne..). Intuitivement, utiliser plusieurs chemins semble offrir une perspective d'amélioration et de garantie, sous réserve bien sûr qu'ils soient les plus disjoints possibles.

Dans ce paragraphe, nous décrivons une approche combinant l'exploitation des chemins multiples avec un mécanisme de contrôle de congestion. Le but recherché est de minimiser le risque de saturation du réseau tout en garantissant le transfert de trafic entre les noeuds communiquants. On espère ainsi améliorer et contrôler le fonctionnement du système en s'appuyant sur les services offerts par la topologie du réseau et la technique de routage multi-chemins utilisée dans les réseaux IP/MPLS.

4.2.4.1 Principe

Afin d'améliorer une telle stratégie, nous proposons de revenir sur la méthode de synthèse et de modélisation du transfert multi-chemins présenté

dans la chapitre 3, notamment la Méthode de Transition Graphique (MTG), et d'y insérer la problématique de contrôle de congestion. Son exploitation consiste à supposer l'existence de plusieurs chemins déterminés par une procédure a priori non précisée. Il convient de définir comment utiliser ces derniers et dans quel ordre. En particulier on cherche à combiner ces chemins avec une méthode de contrôle afin d'éviter la congestion. L'avantage principal ici est d'éviter la congestion, où en cas de saturation, d'autres chemins de substitutions sont déjà connus et prêts à l'utilisation. L'emploi simultané des chemins donne tout son sens à la notion de multi-chemins. L'idée est alors de profiter de cette diversité pour gérer le transfert des flux sur les chemins à disposition dans le réseau. Ici, on ne s'attache pas au choix ou à l'ordre des chemins pour envoyer le premier trafic entre les noeuds communiquant, mais plutôt aux conditions de transfert, notamment la disponibilité de bande passante.

Donc, le principe général est le suivant: le chemin établi pour le transfert de trafic entre une paire source/destination est surveillé, alors que le changement du trajet des flux est supposé s'effectuer lorsque le chemin en cours d'utilisation commence à se saturer (manque de bande passante résiduelle), dans ce cas le trafic est réorienté vers un autre chemin possible entre cette paire source/destination.

En effet, si les flux routés se concentrent systématiquement sur le même chemin, un débordement dans ce cas est fortement possible. Dans certains cas, cet encombrement est structurel, car comme toute entité de communication, un chemin trop fortement sollicité ne pourra gérer la totalité des données qu'il reçoit. A présent, le mécanisme de contrôle aurait tendance à mobiliser d'autres chemins jusqu'à maintenant inutilisés ou non-saturés, ce qui réduit le risque de saturation du chemin et ainsi la congestion du réseau.

4.2.4.2 Fonctionnement

Pour un transfert donné, on suppose qu'une paire source/destination peut acquérir des informations suffisantes sur le réseau afin d'établir une liste de chemins possibles. En pratique, il faut que ces chemins soient disjoints au maximum. On considère que les liens possèdent une capacité unitaire, c'est la valeur maximale d'un flux entre la source et la destination. La technique consiste donc à modifier la mise en oeuvre de l'équation générale du modèle d'état du réseau développé dans le chapitre 3, pour qu'elle puisse intégrer un mécanisme de surveillance et du contrôle de congestion. Ce contrôle intervient dès le premier signe de saturation des chemins du réseau à cause d'une éventuelle surcharge de ses liens. Cependant, du côté surveillance, l'exploitation du réseau nous a permis dans le paragraphe 4.2.2 de calculer à tout instant la bande passante résiduelle d'un chemin. Disposant de cette information, on va pouvoir surveiller la capacité d'un chemin à satisfaire les nouvelles demandes depuis la source de trafic. Par contre, dans le paragraphe 3.3.2.1, l'ensemble des ressources du réseau (topologie, trafic,..) ont été modélisé et une nouvelle méthode pour la synthèse du réseau a été établie: la Méthode de Transition Graphique (MTG). Cette méthode permet de relier les relations: source/destination, chemins, liens et routeurs. La matrice B des relations "sources/destinations et chemins" permet de mette en oeuvre, pour chaque source/destination, la liste des chemins entre cette paire, que le trafic peut emprunter pour assurer le transfert des flux dans le réseau. Modifions cette matrice en la décomposant en deux sous matrices B1 et B2, où B1 est la sous matrice des relations "sources/destination et chemins candidats" et B2 la sous matrice des relations "sources/destinations et chemins sélectionnés", de façon qu'à un moment donné, un seul chemin est actif entre une paire source/destination. En sachant que toutes les autres matrices de

la méthode MTG dérivent de la matrice B2. Dans ce cas, supposons que la source a établi un des chemins de la matrice B1 on modifie donc sa valeur dans la matrice B2 pour la mettre à 1 (c.à.d qu'il est utilisé pour le l'acheminement du trafic), tandis que les valeurs des autres chemins pour la même paire source/destination demeurent à 0. Dans ce cas, en utilisant la matrice B2 pour le calcul des matrices dérivées dans la méthode MTG, seul le chemin sélectionné à 1 est pris en compte lors du transfert du trafic entre cette paire source/destination.

Par exemple, la paire source/destination S/D possède comme chemins candidats l'ensemble (c_1, c_2, c_3), et pour transférer la première demande elle utilise le chemin c_1, alors la matrice B1 aura la forme suivante:

S/D|Chemins candidats	c_1	c_2	c_3	...
s	1	1	1	...
...	.	.	.	.

Et la matrice B2 aura la forme suivante :

S/D|Chemins utilisés	c_1	c_2	c_3	...
s	1	0	0	...
...	.	.	.	.

Pendant ce temps, la bande passante résiduelle du chemin sélectionné est surveillé en permanence, en s'assurant que la condition d'avoir toujours assez de bande passante pour l'envoi de chaque nouvelle demande est satisfaite. Maintenant, si à un moment donné, le chemin sélectionné présente un signe de saturation ou de non satisfaction d'une nouvelle demande, la source choisit un autre chemin de la matrice B1 des chemins candidats et met sa valeur à 1 dans la matrice B2 des chemins sélectionnés. Par ailleurs, l'acheminement de la nouvelle demande sera réalisé en utilisant le nouveau chemin sélectionné.

Le diagramme dans la figure (4.8) montre le principe du modèle de contrôle par chemins:

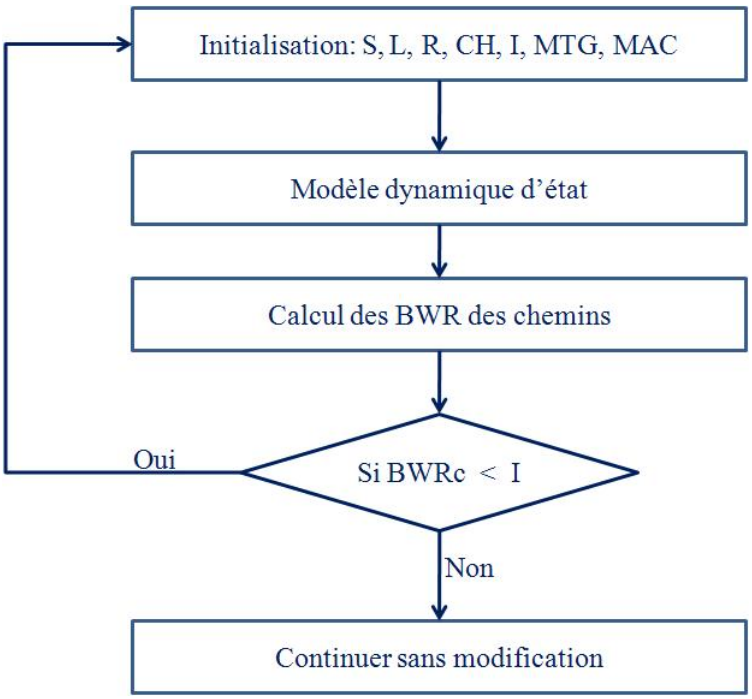


FIG. 4.8 – *Diagramme du modèle de contrôle par chemins*

Avec les notations suivantes: S: ensemble des sources/destinations, L: ensemble des liens, R: ensemble des routeurs, CH: ensemble des chemins, I: ensemble des trafics d'entrée, MAC: Méthode d'Assemblage Comportementale, MTG: Méthode de transition graphique, BWR: Bande Passante Résiduelle d'un chemin.

4.2.4.3 Exemple

L'algorithme proposé répond à l'idée de recherche de chemins non utilisés ou moins utilisés. Néanmoins, notre but est d'éviter de surcharger les chemins et non pas de leur interdire de transporter tout nouveau trafic. De

même, les chemins seront utilisés tant que les variations de leur bande passante résiduelle assurent le transfert du trafic. Cependant, ce mécanisme de contrôle agit au niveau du noeud source du réseau, en toute corrélation avec les réseaux IP/MPLS qui accorde un rôle principale à la source du transfert afin de garantir le choix des chemins.

Dans cette section, nous considérons le même réseau que la partie précédente, avec les mêmes caractéristiques (Nombre de noeuds, taux de service des files d'attentes, etc). Nous choisissons un couple de source/destination S_1/D_1 qui communique entre eux à travers les 2 chemins 1 et 2 comme illustré dans la figure (4.9):

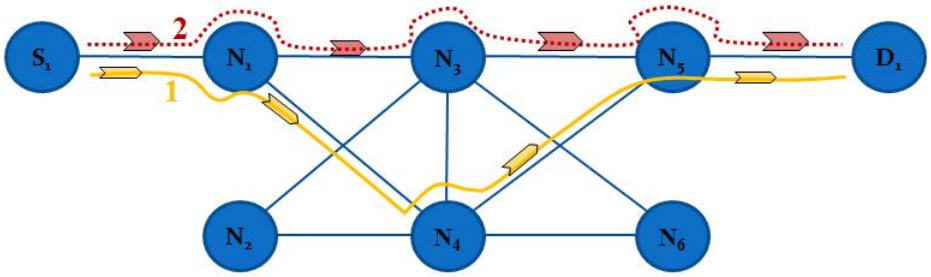


FIG. 4.9 – *Représentation des caractéristiques du réseau*

A l'initialisation, nous supposons que le couple S_1/D_1 utilise le chemin 1 pour transporter le trafic. En répétant les mêmes processus que dans le paragraphe 3.3, on peut établir le modèle dynamique d'état correspondant à ce réseau. Notamment, on s'intéresse à la construction des 2 matrices $B1$ et $B2$.

B1: S/D|Chemins-candidats	1	2
S_1/D_1	1	1

B2 : S/D|Chemins-sélectionnés	1	2
S_1/D_1	1	0

A partir de la matrice $B2$, on peut construire toutes les autres matrices associées pour le développement de la méthode MTG (paragraphe 3.3.2.1). Ainsi avec la méthode MAC (paragraphe 3.3.1.1), on établit le système d'équations d'états du réseau:

$$\begin{aligned} \dot{x}_1 &= I_1 - f_{14} \\ \dot{x}_4 &= f_{14} - f_{45} \\ \dot{x}_5 &= f_{45} - f_{5D_1} \end{aligned} \tag{4.7}$$

En remplaçant les flux de transfert f par leurs valeurs on obtient:

$$\begin{aligned} \dot{x}_1 &= I_1 - u_{14}\ \mu_1 \tfrac{x_1}{(1+x_1)} \\ \dot{x}_4 &= u_{14}\ \mu_1 \tfrac{x_1}{(1+x_1)} - u_{45}\ \mu_4 \tfrac{x_4}{(1+x_4)} \\ \dot{x}_5 &= u_{45}\ \mu_4 \tfrac{x_4}{(1+x_4)} - \ \mu_5 \tfrac{x_5}{(1+x_5)} \end{aligned} \tag{4.8}$$

Cependant le calcul des bandes passantes résiduelles des chemins (dans ce cas il s'agit du seul chemin 1), comme indiqué dans le paragraphe 4.2.2, ainsi que sa surveillance sont faits conjointement pendant l'évolution du modèle dynamique d'état. Si la valeur de la bande passante résiduelle atteint le seuil limite, le contrôleur réagit automatiquement en déviant la demande vers le chemin candidat non utilisé, à savoir dans ce cas le chemin 1. Cette déviation est faite en remplaçant les valeurs de la matrice $B2$, et ainsi le modèle dynamique est établi avec les nouveaux paramètres configurés.

Pour la simulation, les taux de services μ_i, $i = 1..6$, sont fixés à 10, la demande pour la paire source/destination S_1/D_1 est fixée à 4, et les bandes passantes de deux chemins 1 et 2 sont fixées à 10. Ce qui revient à dire qu'à l'initialisation, les valeurs des bandes passantes résiduelles des chemins 1 et 2 sont égales à 10. Nous pouvons observer le phénomène de transfert de la demande entre la paire S_1/D_1 sur la figure (4.10). Au départ, on remarque que la bande passante résiduelle du chemin 1 ($BWR1$) commence à diminuer

au fur et à mesure que la demande arrive, tandis que celle du chemin 2 ne bouge pas et reste à 10, vu qu'au début, seul le chemin 1 est sélectionné pour acheminer le trafic entre la paire S_1/D_1. Lorsque la bande passante résiduelle du chemin 1 atteint le seuil de saturation, à partir du moment où le reste de sa bande passante ne devient plus satisfaisant pour assurer la demande, alors le trafic est orienté vers le chemin 2. On peut remarquer cette déviation de trafic sur la figure (4.10) par la diminution de la bande passante résiduelle du chemin 2. Ainsi, le trafic est assuré entre la paire source/destination en évitant la congestion du réseau.

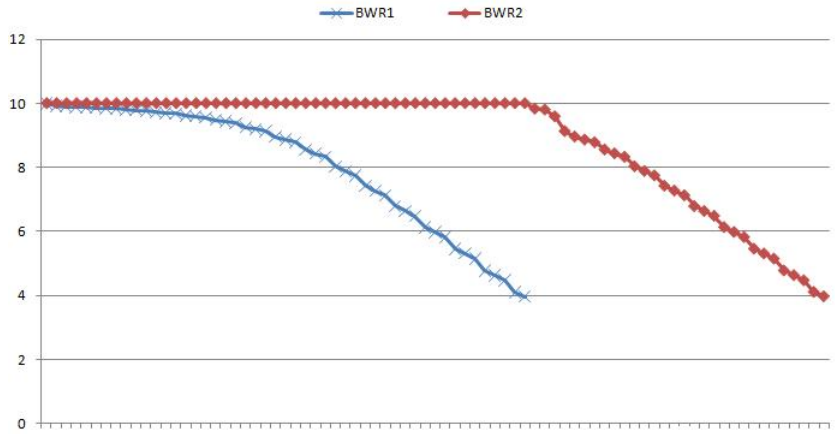


FIG. 4.10 – *Evolution des bandes passantes résiduelles des chemins 1 et 2*

4.3 Conclusion

Dans ce chapitre, nous avons entamé une étude sur les mécanismes de contrôle dans un réseau de type IP/MPLS basé sur le modèle analytique présenté dans le chapitre précédent. Nous avons fixé un objectif de proposer et valider plusieurs modèles de contrôle. Ces modèles originaux, représentent des moyens pour mieux gérer les ressources du réseau et lui éviter une situation

de saturation ou de congestion.

Dans un premier temps, nous avons formulé un modèle de contrôle en rétroaction basé sur l'information de la capacité de transport des chemins du réseau. En effet, le fait de disposer de modèles basés sur des acheminements de bout-en-bout et d'autres sur des routages de saut-en-saut, nous a inspiré un modèle qui représente une ligne complète de l'émetteur au récepteur en passant par les liens du réseau. En plus, afin de tenir compte de l'influence de l'évolution des paramètres essentiels du réseau, notamment la bande passante des chemins, un environnement extérieur est ajouté au modèle analytique principal. Ensuite, une entrée supplémentaire est introduite au chemin d'entrée afin de contrôler la demande selon la capacité du réseau. Cette entrée possède un profil variable entre 0 et 1, à l'image de ce qui peut se produire sur les chemins du réseau et notamment leurs états de charge. Les résultats ont montré une déduction a priori à chaque intervalle de temps des demandes d'entrée selon la capacité des chemins du réseau.

Dans un second temps, nous avons présenté un modèle de contrôle original basé sur le routage multi-chemins. Nous avons pu intégrer ce modèle aussi au modèle analytique, notamment à la partie de représentation et de modélisation du modèle d'état global, à savoir la Méthode de Transition Graphique (MTG) définie dans le chapitre précédent. Ce modèle consiste à réorienter la demande de trafic à chaque passage dans une période de saturation des chemins, et ainsi éviter la congestion du réseau. Ensuite, grâce aux résultats de simulation, nous avons observé comment le trafic a dévié automatiquement à l'instant de saturation.

Le modèle de contrôle peut être généralisé en l'intégrant à la partie de construction du modèle d'état global, par la création de matrices spécifiques correspondantes aux variables de contrôle z et β. De même, le modèle de

contrôle par routage multi-chemins peut être généralisé et reproduit à chaque passage d'un chemin à l'autre.

Conclusions et Perspectives

Au terme de ce manuscrit, cette partie constitue un récapitulatif de notre travail, ainsi qu'une analyse globale des résultats obtenus afin de dresser des perspectives qui permettront d'envisager de nouveaux axes de recherche comme une suite logique de ce travail. Ce travail de thèse était double, d'une part l'évaluation de nouveaux modèles pour l'ingénierie de trafic, et d'autre part de proposer un modèle d'état pour les réseaux IP/MPLS.

Conclusions

Nous avons traité tout au long de ce travail de thèse les réseaux de communication de type IP/MPLS, avec des mécanismes d'ingénierie de trafic et de qualité de service DiffServ. Notre contribution a porté sur trois parties.

Dans la première partie, nous nous sommes intéressés à l'évaluation des performances des modèles de routage multi-chemins pour faire de l'ingénierie de trafic dans les réseaux IP/MPLS, à savoir LBWDP et PEMS qui ont été développés dans [59][58][56]. Après un rappel de quelques notions sur les réseaux IP/MPLS, nous avons effectué une étude comparative des générateurs des topologies et retenu le générateur BRITE, qui permet de générer des topologies semblables à Internet. Nous avons développé un script pour transformer la topologie générée par BRITE vers un fichier utilisable par le Simu-

lateur Réseau NS-2, la plateforme de simulation. Nous avons également défini les indices permettant d'évaluer les performances de deux modèles LBWDP et PEMS, notamment leur capacité à équilibrer la charge du réseau. Ainsi, nous avons proposé une méthode de simulation composée de trois étapes à savoir la topologie, le trafic et le modèle. Cette méthode a permis d'obtenir et d'analyser les indices de performances de ces modèles sur des topologies ressemblant à Internet en taille et complexité et selon différents scénarios de charge du réseau. Nous avons analysé la scalablité des modèles en augmentant la taille et la complexité du réseau. Nous avons remarqué que l'équilibrage obtenu dans le cas d'un réseau de grande taille est meilleur que celui obtenu pour un réseau de petite taille. D'autre part, nous avons constaté que l'état de charge du réseau influence fortement les performances des deux modèles. Leur comportement a changé d'un scénario à l'autre. Enfin, nous avons montré, d'après des simulations sur différentes topologies générées par BRITE, que la structure du réseau agit faiblement sur les performances du modèle.

A partir de ces différentes remarques, nous avons conçu un modèle dynamique d'état pour les réseaux de type IP/MPLS dans la deuxième partie. Ce modèle est basé sur le principe de routage multi-chemins. Il permet de calculer non seulement la charge du réseau, mais aussi les flux de transfert de trafic sur les chemins du réseau. Il est établi sur la base de la théorie différentielle du trafic. Il est capable de développer les équations dynamiques régissant le comportement dynamique du réseau. Celui-ci est orienté tout particulièrement vers la technologie MPLS et notamment la technique du routage explicite, mais il pourrait facilement être étendu à d'autres types de réseaux. Il repose sur deux approches de représentation, une comportementale et l'autre graphique, afin de rendre possible la modélisation d'une façon générale du réseau. L'intérêt de ce modèle réside dans l'exploitation

dynamique des informations et des mesures obtenues lors du transfert du trafic dans le réseau, afin de quantifier les paramètres caractérisant un goulet d'étranglement dans les chemins du réseau, le tout en fonction du plan de routage adopté.

Enfin, l'un des objectifs de cette thèse a été d'étudier le problème de congestion et de proposer des solutions d'équilibrage de charge. Pour cela, la troisième partie de ce travail a été consacrée à l'étude des modèles de contrôle de congestion. En effet, l'étude du modèle dynamique d'état ainsi construit nous a permis de développer des mécanismes de contrôle de congestion dans le réseau IP/MPLS. Nous avons pu constater d'après les conclusions de la première partie, que la politique de contrôle de congestion pourrait en effet offrir une meilleure qualité de service, ainsi qu'une meilleure gestion des ressources du réseau. Ainsi, nous avons pu construire deux modèles de contrôle de congestion adaptées aux réseaux IP/MPLS. Comme nous avons constaté que la QdS d'un flux peut être détériorée à cause d'une charge élevée sur au moins un des liens qui forment le chemin de bout en bout. Ces modèles conçoivent une méthode d'adaptation dynamique de trafic en fonction des capacités disponibles des chemins du réseau. Notre contribution a porté sur l'intégration du concept de l'ingénierie de trafic par équilibrage de charge pour assurer le transfert des flux de trafic, tout en évitant la congestion dans le réseau. Le premier modèle de contrôle permet de ralentir momentanément le trafic et le moduler à des valeurs inférieures à la demande, tandis que le deuxième modèle offre au réseau la capacité de réorienter le trafic d'un chemin sur ceux qui ne sont pas congestionnés. Ces deux modèles peuvent être facilement déployés avec le modèle dynamique d'état proposé.

Perspectives

Les travaux menés tout au long de cette thèse ainsi que les résultats obtenus permettent de dégager plusieurs perspectives directement liées à l'évaluation de performances des deux modèles LBWDP et PEMS, et au modèle adaptatif pour la qualité de service dans les réseaux IP/MPLS.

Première partie

Cette partie est consacrée à l'étude des perspectives des deux modèles LBWDP et PEMS. En interprétant les résultats des simulations, on a conclu que l'état de charge du réseau agit sur les performances de ces deux modèles. On a remarqué que leur comportement a changé d'un scénario à l'autre, tout en changeant les performances des paramètres de qualité de service visés. Ce changement de performances nous amène donc à développer d'autres techniques issues du comportement de ces modèles ou de leurs caractéristiques. Dans un premier temps, nous mettons en oeuvre une approche de commande multi-modèles selon l'état de charge du réseau. Puis, nous allons définir les bases d'une nouvelle approche de routage multi-chemins, dont laquelle le choix des critères de sélection des chemins prend en compte les analyses des résultats obtenus lors de l'étape de l'évaluation de performances.

Commande mulit-modèles sur l'entrée

Cette approche agit à l'entrée d'un réseau FAI. Son principe consiste à équilibrer les performances des deux modèles, en choisissant à chaque demande de trafic entrant, le modèle à appliquer pour acheminer ce trafic dans le réseau. En effet, l'idée vient du fait que les performances des deux modèles changent selon le scénario de trafic appliqué. Cependant, nous avons pu re-

marquer que si la demande du réseau est élevée, le modèle LBWDP semble meilleur que PEMS. PEMS possède des performances plus adaptées à un réseau avec une faible demande. L'idée est donc d'imposer un test de contrôle du volume de la demande à l'entrée du réseau. Ensuite, selon le résultat de la comparaison du volume de trafic d'entrée avec un seuil fixé, et qui dépend de l'état du réseau, on choisit le modèle à utiliser pour l'acheminement de la demande.

Par exemple, lors des dernières simulations, nous avons estimé qu'une faible demande à 100[KB] correspond bien à un réseau à faible charge. Par contre, quand la demande dépasse le seuil de 100 [KB], les liens du réseau deviennent de plus en plus chargés et le risque de congestion devient plus important. Nous allons donc fixer le seuil de comparaison à 100 [KB]. Ainsi lors d'un même scénario, les demandes sont générées aléatoirement depuis les sources, avec un volume aléatoire entre 1 [Kb] et 1000 [KB] par exemple, correspondant à plusieurs états de charge. Ensuite, à chaque demande générée, on procède au test de contrôle entre le volume de cette demande et le seuil fixé. Si le volume de la demande est plus grand que le seuil (100[KB]), on utilise LBWDP, et s'il est plus petit que le seuil, alors PEMS sera utilisé.

Cette approche présente des avantages au niveau de l'équilibrage de charge et le contrôle de congestion dans le réseau, car elle permet de choisir le modèle à appliquer à chaque fois qu'une demande arrive au réseau. L'inconvénient de cette approche réside au niveau de la complexité d'implémentation, comme par exemple la sélection des chemins candidats à chaque fois qu'une demande arrive. Cela peut être résolu en insérant une étape hors ligne de calcul des chemins candidats pour chaque modèle et d'enregistrement de cet ensemble.

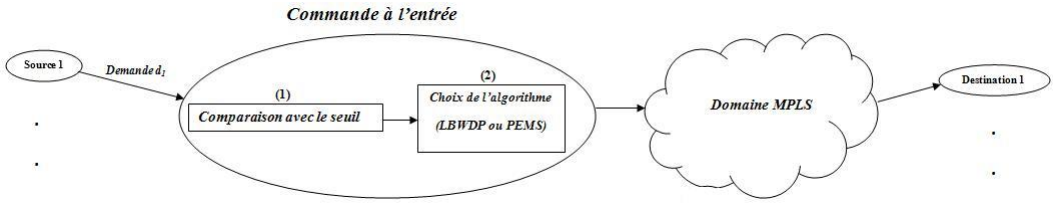


FIG. 4.11 – *Principe général du modèle de commande multi-modèles*

Proposition d'une nouvelle approche

Certes les 2 modèles LBWDP et PEMS ont été établi pour faire de l'ingénierie de trafic dans les réseaux IP/MPLS, et certes ils utilisent les mêmes métriques pour la qualité de service, à savoir la bande passante et le délai, mais leur fonctionnement diffère dans la manière d'implémentation, et la manière d'établissement des chemins candidats. Cette différence nous incite à développer une approche combinant les points forts de chacun d'eux. Nous avons remarqué que PEMS est sensible à l'état de charge du réseau. Il fonctionne bien avec un réseau à faible charge. Par contre, dans un réseau à charge élevée, PEMS continue à différencier le trafic selon chaque classe, mais au détriment de l'équilibrage de charge, où dans ce scénario, LBWDP a montré des performances meilleurs que celles de PEMS. Cependant, l'équilibrage de charge est une métrique très importante pour la mesure des performances d'un réseau, car un bon équilibrage de charge diminue le risque de congestion du réseau et ainsi la perte de paquets. Cette métrique essentielle est proportionnelle au critère de la bande passante résiduelle des chemins du réseau. Plus l'équilibrage est meilleur, plus les chemins possèdent des bandes passantes résiduelles et inversement. C'est grâce à ce rapport entre l'équilibrage et la bande passante que LBWDP possède les meilleurs performances dans le cas d'un réseau chargé, car dans son implémentation, LBWDP prend en considération la largeur de la bande passante comme critère principal lors de

la procédure de sélection des chemins candidats. Ainsi, les chemins choisis à chaque mise à jour, possèdent les meilleures bandes passantes résiduelles, ce qui n'est pas forcément le cas de PEMS, où entre les différentes étapes de prétraitement, sélection de chemins candidats et ensuite l'affectation de chemins par classe de trafic, la mise à jour ne lui permet pas nécessairement de trouver les meilleurs chemins en terme de bande passante résiduelle, surtout si le nombre de chemins par classe est limité. Par ailleurs, les résultats ont montré une corrélation entre les utilisations des chemins et les délais. Ce qui paraît logique, vu que le délai composé entre autre du délai de transmission dépendant de la bande passante du chemin et du délai d'attente dépendant du stockage dans les files d'attente. Ces facteurs ne font qu'augmenter le délai au cas où l'équilibrage est faible pour un modèle ou un autre. Par conséquence, l'idée est de grader les points forts de chaque modèle et de proposer un nouveau modèle qui les assemble. Du côté LBWDP, son point fort est le bon équilibrage de charge dû au critère de la largeur d'un chemin, par contre côté PEMS, son point fort réside dans sa capacité à différencier le délai selon les classes de trafic.

Une proposition consiste à améliorer PEMS en modifiant sa structure de la manière suivante:

1- Enlever l'étape de prétraitement

En effet l'intérêt même qu'elle s'effectue hors ligne, présente aussi plusieurs inconvénients. Premièrement, elle se base sur le critère de plus courts chemins en nombre de sauts, ce qui peut être intéressant si c'était une mesure de la bande passante. Deuxièmement, elle offre un nombre limité de chemins, et qui ne sera pas mis à jour lors de l'étape de sélections des chemins candidats. D'où, dans un scénario ou des nombreux trafics élevés, les chemins limités sont rapidement saturés, ce qui provoque une congestion du réseau, ainsi un

mauvais équilibrage de charge et un délai élevé.

2- Reprendre la part de sélection des chemins candidats de LBWDP, où le critère de bande passante résiduelle est essentiel, et ensuite l'intégrer avec l'étape de choix des chemins par classe de trafic. Dans ce cas, la mise à jour s'effectuera à cette période.

3- Garder l'étape de répartition de trafic de PEMS.

Cette approche offre l'intégration de la largeur d'un chemin et la différenciation de délai qui sont les deux points forts de LBWDP et de PEMS, et assure un fonctionnement dans tous les scénarios de charge possible. Ainsi, son implémentation ne devrait pas présenter une complexité plus forte que celle du PEMS ou de LBWDP.

Deuxième partie

Cette partie présente les perspectives liées au modèle dynamique d'état développé dans le chapitre 3 ainsi qu'aux deux modèles de contrôle proposés dans le chapitre 4.

Le modèle dynamique représente l'état du réseau. Il prend en compte les paramètres de la topologie du réseau, ainsi que de trafic d'entrée, et génère un système d'équations qui permet de mettre en oeuvre les interactions entre les différents éléments du réseau. Son développement est quasi complet, au moment où l'utilisateur précise les chemins à emprunter par le trafic entrant au niveau des sources. Donc, comme mentionnée dans les hypothèses de chapitre 3 lors de l'étude du modèle, le choix des chemins entre une paire source/destination est une simple hypothèse et il n'est pas basé sur des critères quelconques comme pourrait être dans certains cas. En effet, une telle perspective de sélection des chemins constitue un prétraitement de l'étape

de modélisation, dont les hypothèses sur le choix des critères de sélection peuvent varier selon le cas étudié. D'une manière générale, on peut considérer des critères simples pour l'étape de prétraitement comme la longueur d'un chemin en nombre de sauts ou la minimum disjonction possible. La réalisation de cette perspective peut s'inspirer de l'approche graphique ainsi que de la méthode de Transition Graphique développé dans le chapitre 3. Sa complexité ne devrait pas être pénalisant, vue que d'une part ce genre d'étapes est souvent effectué hors ligne, et d'autre part de la nature de l'approche graphique qui est basée sur des matrices d'incidences de 0 et 1 de très faible complexité calculatoire . Notons que ces critères peuvent être étendus vers d'autres paramètres de qualité de service comme le délai, le débit, etc., où la sélection des chemins peut être régénérée à chaque mise à jour dynamique en fonction de l'état de charge du réseau, devenu réalisable grâce au modèle dynamique d'état ainsi développé.

En ce qui concerne les modèles de contrôle, leurs perspectives se concentrent dans le fait de généralisation sur un réseau entier. Par ailleurs, le modèle de contrôle par rétroaction peut être généralisé avec le modèle dynamique du réseau, de façon qu'une fois le système d'équations du modèle dynamique généré, celui du modèle de contrôle sera généré automatiquement et d'une manière générale avec les valeurs des matrices de trafic entrant et sortant d'un routeur ainsi que les valeurs des variables de routage correspondants, afin de déterminer le vecteur d'état de ce modèle. Le modèle de contrôle multi-chemins peut aussi être adapté d'une manière générale avec le modèle dynamique d'état et étudié sur un réseau entier avec plusieurs sources de trafic, ainsi que le choix du premier chemin à emprunter pour acheminer le trafic, qui avec la perspective de l'étape de sélection des chemins, sera le meilleur chemin dans l'ensemble des chemins choisis.

Bibliographie

[1] Andersson L., Minei I., "LDP Specification", IETF, RFC 5036, 2007.

[2] ATM Forum, Private Network-Network Interface Specification Version, 1996.

[3] Awduche D., Berger L., Gan D., LI T., Srinivasan S., Swallow G., RSVP-TE: Extensions to RSVP for LSP Tunnels, Cisco Systems, Inc., 2001.

[4] Awduche D., Chiu A., Elwalid A., Widjaja I., and Xiao X., Overview and Principles of Internet Traffic Engineering, IETF RFC 3272, 2002.

[5] Bachar T. , Altman E., Srikant ER, "Robust Rate Control for ABR Sources", IEEE conference on Decision and Control, San Diego, CA, 1998.

[6] Bachar T.,Altman E., Srikant ER, "Flow control in communication networks with miltiple users and action delays: A team theoretic approach", 1997.

[7] Bastin G., Guffens V., "Congestion control in compartmental network systems", CESAME, Université Catholique de Louvain, ELSEVIER, 2006.

[8] BASTIN G., "Sur la modélisation et le contrôle des réseaux dynamiques conservatifs", CESAME, Université Catholique de Louvain, CIFA 2006.

[9] BOCKSTAL Ch. "Modélisation différentielle du trafic et simulation hybride de réseaux IP-MPLS DiffServ", Thèse de Doctorat, LAAS-CNRS,

2005.

[10] Boyan J.A. and Littman M.L., "Packet Routing in Dynamically Changing Networks: A Reinforcement Learning Approach ", Cowan, Tesauro and Alspector (eds), Advances in Neural Information Processing Systems 6, 1994.

[11] Braden R., Zhang E.L., Berson S., Herzog S., Jamin S., "Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification", http://www.faqs.org/rfcs/rfc2205.html, 1997.

[12] Bradley T., Brown C., Malis A., "Multiprotocol Interconnect over Frame Relay", IETF, RFC 1490, 1993.

[13] BRITE, "BRITE : Boston university Representative Internet Topology gEnerator", Computer Science Department Boston University, BUCS-TR-2001-003, 2001.

[14] BRUN O., "Modélisation et optimisation de la gestion des ressources dans les architectures distribuées-Parallélisation massive de la technique de filtrage particulaire", Thèse de Doctorat, LAAS-CNRS, 2000.

[15] Callon R., Using of OSI IS -IS for routing in TCP-IP and Dual Environments, IETF RFC 1195, 1990.

[16] Calvert K., Doar M., and Zegura E., "Modeling Internet Topology",IEEE Transactions on Communications, pages 160-163, 1997.

[17] CASELLAS R., "Partage de Charge et Ingénierie de Trafic dans les Réseaux MPLS", Thèse de Doctorat, Telecoms Paris, 2002.

[18] Chaieb I., "Ingénierie de Traffic avec MPLS : Routage Distribué", Thèse de Doctorat, IRISA, Rennes, 2007.

[19] Chao X., Miyazawa M., Pinedo M., "Queueing Networks: Customers, Signals and Product Form Solutions", John Wiley and Sons, 1999.

[20] CISCO, An Introduction to IGRP, Document ID: 26825, 1991.

[21] Claffyy K.C and McRobb D., "Measurement and Visualization of Internet Connectivity and Performance", http ://www.caida.org/Tools/Skitter.

[22] Conseil Scientifique France Télécom. L'évolution des Réseaux. Forum France Télécom recherche, Mémento technique N°15, ISSN March 1250-5447, 2000.

[23] Crawley E., Nair R., Rajagopalan b., Sandick H., "A Framework for QoS-based Routing in the Internet", RFC2386, IETF, August 1998.

[24] Davie B., Charny A., Bennett J.C.R., Benson K., Le Boudec J.Y., Courtney W., Davari V., Firoiu V. et Stiliadis D., "An Expedited Forwarding PHB (Per-Hop Behavior)", IETF RFC 3246, 2002.

[25] Differentiated Services (DiffServ), IETF, RFC 2475, 2003.

[26] Doar M., "A Better Model for Generating Test Networks", In Proceeding of IEEE GLOBECOM, November 1996.

[27] ECMP: Equal-Cost Multi-Path Algorithm, RFC 2992, http://www.ietf.org/rfc/rfc2992.txt.

[28] Elwalid A. , Low S. and Widjaja i. , "MATE: MPLS Adaptive Traffic Engineering", INFOCOM, 2009, 1300-1309.

[29] Faloutsos M., "On power law rrelationships of the internet topology", ACM SIGCOMM 99, pages 256-262, Cambridge, USA, 1999.

[30] Feldmann A., Greenberg A., Lund C., Reingold N., Rexford J., "NetScope : Traffic engineering for IP Networks", 1999.

[31] Fortz B., Thorup M., "Internet Traffic Engineering by Optimizing OSPF Weights", 2000.

[32] Fortz B., Thorup M., "Internet Traffic Engineering by Optimizing OSPF Weights", INFOCOM, 2000, (2), 519-528,.

[33] Fortz B., Thouryp M., "Optimizing OSPF/IS-IS weights in a changing world", Selected Areas in Communications, IEEE Journal, volume (20), 2002.

[34] Garcia J.M., " Problèmes liés à la modélisation du traffic et à lâcheminement des appels dans un réseau téléphonique", Thèse de doctorat, université Paul sabatier, Toulouse, 1980.

[35] Garcia J.M., "A new approach for analytical modelling of packet switched telecommunication networks", LAAS-CNRS Research Report No98443, 1998.

[36] GARCIA J.M., GUACHARD D., BRUN O., BACQUET P., SEXTON J., LAWLESS E., " Modélisation différenticielle du trafic et simulation hybride distribuée", Rapport LAAS 01660, 2001.

[37] Garcia J.M., Legall F., Bernussou J., "A Model for Telephone Networks and its Use for Routing Optimization Purposes", IEEE Jour. on Select. Areas in Comm., Special issue on communication network performance evaluation, Vol. 4, 1986.

[38] GAUCHARD David, "Simulation Hybride des Réseaux IP-DiffServ-MPLS Multi-services sur Environnement d'Exécution Distribuée", Thèse de Doctorat, LAAS-CNRS, 2005.

[39] Gorunescu M., Gorunescu F., "Modeling the kinetics behind the patients?ow", AMS subject classi?cation: 90B10, 93A30, 34A50, 2007.

[40] Gross C.M., Harris D., "Fundamentals of Queueing Theory", John Wiley & Sons, 1998.

[41] Guerin R., Orda A.,Williams D., "Quality of Service Routing Mechanisms and OSPF Extensions", IETF Internet Draft, 1996.

[42] Haddad W., Chellaboina V., Hui Q., "Nonnegative and compartmental dynamical systems", Princeton University Press, ISBN 978-0-691-14411-

5.

[43] Hayakawa T., "Compartmental Modeling and Neural Network Control of Stochastic Nonnegative Systems", IEEEXplore, 2007.

[44] HAYAKAWA Y, Shigeyouki H, MUTSUMi H, MASAMI I , "On the structural controllability of compartmental systems", IEEE transactions on Automatic Control, Vol; AC 29, 1986.

[45] Hedrick C., "Routing Information Protocol", RFC 1058, 1988.

[46] Heinanen J., Baker F., Weiss W. et Wroclawski J., "Assured Forwarding PHB Group", IETF RFC 2597, 1999.

[47] Higashi M., "Residence time in constant compartmental ecosystems", Ecological Modelling , Volume 32, Issue 4, July 1986, Pages 243-250.

[48] IETF, The Internet Engineering Task Force, http://www.ietf.org/.

[49] Information Sciences Institute « Transmission Control Protocol » University of Southern California, 1981 http://www.faqs.org/rfcs/rfc793.html.

[50] Integrated Services (intserv), IETF, RFC 1633, 2001.

[51] IP, Internet Protocol, IETF, RFC 791, 1981.

[52] Jamoussi B., CR-LDP specification, IETF, RFC 3212, 2002.

[53] Jin C., Chen Q., and Jamin S., "Inet : Internet Topology Generator", Technical Report Research Report CSE-TR-433-00, University of Michigan at Ann Arbor, 2000.

[54] Kelly F., "Models for a self-managed Internet", Centre for Mathematical Sciences, University of Cambridge, Wilberforce Road, Cambridge CB3 0WB, UK.

[55] LAGIS, Laboratoire d'Automatique, Génie Informatique et Signal , Ecole Centrale de Lille, http://lagis.ec-lille.fr/.

[56] Lee K., "Global QoS model in the ISP networks : DiffServ aware MPLS Traffic Engineering", LAGIS, Ecole centrale de Lille, 2006.

[57] Lee K., Rahmani A. , Toguyeni A., "Hybrid Multipath Routing Algorithms for Load Balancing in MPLS Based IP Network", AINA 2006, Austria.

[58] Lee K., Rahmani A., Toguyeni A., "PEMS, a PEriodic Multi-Step routing algorithm for DS-TE", IeCCS'06, 2006.

[59] Lee K., Rahmani A., Toguyeni A., "Stable load balancing algorithm in MPLS network", IMAACA'2005,France.

[60] Lee K., Toguyeni A., Rahmani A. ,"Comparison of multipath algorithms for load balancing in a MPLS Network", ICOIN'2005, Korea.

[61] Lougheed K., Rekhter Y., "A Border Gateway Protocol 3 (BGP-3)", RFC 1267, 1991.

[62] Lucio G.F., Paredes-Farrera M., Jammeh E., Fleury M., Reed M.J., "OPNET Modeler and Ns-2: Comparing the Accuracy Of Network Simulators for Packet-Level Analysis using a Network Testbed", University of Essex, UK.

[63] Malkin G., "RIP version2: Carrying Additional Information", RFC 1388, 1993.

[64] Mascolo S. , "Modeling the Internet congestion control using a Smith controller with input shaping", Dipartimento di Elettrotecnica ed Elettronica, Politecnico di Bari, 70125 Bari, Italy, 2005.

[65] Mascolo S., "Congestion control in high-speed communication networks using the Smith principle", Dipartimento di Elettrotecnica ed Elettronica, Politecnico di Bari, Via Orabona 4, 70125 Bari, Italy, 1999.

[66] Mascolo S., "Smiths Principle for Congestion Control in High Speed ATM Networks", Dipartimento di Elettrotecnica ed Elettronica, Poli-

tecnico di Bari, Via Orabona 4, 70125 Bari, Italy.

[67] The MathWorks Inc, http://www.mathworks.com/.

[68] McDysan D. and Spohn L., "ATM Theory and Application", Mc Graw Hill, 1994,.

[69] Mills D.L., "Exterior Gateway Protocol Formal Specification", RFC 904, 1984.

[70] MNS, "MPLS Simulation Tools", http ://folk.uio.no/johanmp/mpls.html.

[71] Moy J., OSPF Version 2, IETF RFC 1247, 1991.

[72] "Multi-Protocol Label Switching (MPLS) Support of Differentiated Services", IETF RFC 3270, 2002.

[73] Nam, "Nam : Network Animator", http ://www.isi.edu/nsnam/nam/.

[74] Nelakuditi S. and Zhang Z.,"On Selection of Paths for Multipath Routing", University of Minnesota, Deparment of Computer Science Engineering, 2002.

[75] Nichols K., Blake S., Baker F. et Black D., "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", IETF RFC 2474, December 1998.

[76] NLANR, "National Laboratory for Applied Network Research (NLANR)", http ://moat.nlanr.net/rawdata/.

[77] NS2, "The Network Simulator (NS2)", http ://www.isi.edu/nsnam/ns/.

[78] Nucci A., Schroeder B., Bhattacharyya S., Taft N., Diot N., "IGP Link Weight Assignment for Transient Link Failures" International Teletraffic Congress (ITC), 2003.

[79] OPNET Technologies, The OPNET modeler, http://www.opnet.com.

[80] Perez M., Liaw F., Grossman D., ATM Signaling Support for IP over ATM, IETF, RFC 1755.

[81] RACHDI M.A., "Optimisation des ressources de réseaux hétérogènes avec coeur de réseau MPLS", Thèse de Doctorat, LAAS, 2007.

[82] Raj Jain. The art of computer Systems Performance Analysis. John Wiley & Sons, INc, 1991. ISBN: 0-471-50336-3.

[83] Rodrigues M.A. and Ramakrishnan K.G., "Optimal Routing in Shortest Path Networks", ITS, 1994.

[84] Rosen E. and Rekhter Y., BGP/MPLS VPNs, IETF RFC 2547, 1999.

[85] Rosen E., Viswanathan A., Callon R., "Multiprotocol Label Switching Architecture", IETF RFC 3031, 2001.

[86] Simpson W., "PPP in Frame Relay", 1996.

[87] Song J., Kim S. and Lee M.,"Dynamic Load Distribution in MPLS Networks",Department of Computer Science, Ewha Womans University, 2003.

[88] Stevens W., "TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms", 1997, http://www.faqs.org/rfcs/rfc2001.html.

[89] Tcl Developer Site, http://www.tcl.tk/.

[90] Wang Z., Crowcroft J., "Bandwidth-Delay Based Routing Algorithms", IEEE GlobeCom 95, Singapore, Nov 1995.

[91] Waxman B., "GRouting of Multipoint Connections", IEEE J.Select. Areas Commun., December 1988.

[92] Zhang L., "Virtual clock: a new traffic control algorithm for packet switching networks ", ACM SIGCOMM Computer Communication Review, Volume 20 Issue 4, 1990.

Annexes

Algorithme 1 *Algorithme de la Méthode de Transition Graphique*

Entrées: G, R, L, SD, CH, I

Sorties: E, T, G, H

**** Sous procédure de calcul de T ****

B:

1. *Pour* $s \in SD$, $c \in CH$,
2. *Si (s est servi par c) alors* $B_{sc} = 1$,
3. *Sinon* $B_{sc} = 0$

C:

4. *Pour* $c \in CH$, $r \in R$,
5. *Si* $r \in c$, *alors* $C_{cr} = 1$,
6. *Sinon* $C_{cr} = 0$

E:

7. *Pour* $s \in SD$,
8. *Si* $B_{sc} = 1$
9. *Pour* $c \in CH$,
10. *Si* $C_{cr} = 1$ *alors* $E_{sr} = 1$,
11. $T \leftarrow r$

12. *Sinon* $E_{sr} = 0$

**** Sous procédure de calcul de G et H ****

13. *Pour* $c \in CH$, $r \in R$,

14. *Si* $C_{cr} = 1$

15. *Pour* $j \in R$, *t.q* $j \prec r$,

16. *Si* $C_{cj} = 1$ *alors* $G_{rj} = 1$ */** Liens Entrants**/*

17. *Sinon* $G_{rj} = 0$

18. *Pour* $j \succ r$, $G_{r}j = 0$

19. *Pour* $k \in R$, *t.q* $k \succ r$,

20. *Si* $C_{ck} = 1$ *alors* $H_{rk} = 1$ */** Liens Sortants**/*

21. *Sinon* $H_{rk} = 0$

22. *Pour* $k \prec r$, $H_{rk} = 0$

23. *Si* $r = 1$ *alors* $G_{rj} = 0$ *pour tout* $j \in R$ */*premier*/*

24. *Si* $r = n$ *alors* $H_{rk} = 0$ *pour tout* $k \in R$ */*dernier*/*

Titre : Conception et évaluation d'un modèle adaptatif pour la qualité de service dans les réseaux MPLS

L'objectif de ce travail de thèse dans un premier temps est l'évaluation de performances des modèles de routage multi-chemins pour l'ingénierie de trafic et l'équilibrage de charge sur un réseau de type IP/MPLS (MPLS-TE). Nous comparons la capacité de ces modèles à équilibrer la charge du réseau tout en faisant de la différentiation de trafic. Nous les appliquons sur des grandes topologies générées par le générateur automatique des topologies BRITE, qui s'approchent en forme et en complexité du réseau réel. Nous mesurons ainsi l'impact de leur complexité respective et donc la capacité à les déployer sur des réseaux de grande taille (scalabilité). Dans un second temps, l'objectif est de proposer un concept de modélisation générale d'un réseau à commutations par paquets. Ce modèle est établi sur la base de la théorie différentielle de trafic et la théorie des files d'attente, tout en utilisant des approches graphiques. Le but est d'estimer l'état de charge du réseau et de ses composants (routeurs, liens, chemins). Ensuite, en fonction de ça, nous développons des approches de contrôle de congestion et commande sur l'entrée améliorant les techniques de routage adaptatif et l'équilibrage de charge dans les réseaux IP/MPLS.

Mots clés : Réseaux IP-MPLS, Ingénierie de trafic, Qualité de service (QdS), Modélisation Analytique, Contrôle, Commande, Simulations, Evaluation de performances, Passage à l'échelle.