THESE DE DOCTORAT CONJOINT TELECOM SUDPARIS et L'UNIVERSITE PIERRE ET MARIE CURIE

Spécialité :

Ecole doctorale : Informatique, Télécommunications et Electronique de Paris

Présentée par

LE TIEN ANH

Pour obtenir le grade de
DOCTEUR DE TELECOM SUDPARIS

APPROCHE CROSS-LAYER POUR SERVICES MULTIMEDIA ÉVOLUTIFS DISTRIBUÉS SUR LA PROCHAINE GÉNÉRATION DE RÉSEAUX CONVERGENTS : ARCHITECTURES ET PERFORMANCES

Soutenue le 15/06/2012

devant le jury composé de :

M. NOEL CRESPI, Directeur d'Etude, Hdr, Telecom SudParis	**Directeur de thèse**
Mme. HANG NGUYEN, Directeur d'Etude, Telecom SudParis	**Encadrant de these**
M. ANTHONY BUSSON, Hdr, Supélec	**Rapporteur**
M. SAMIR TOHME, Prof, Hdr, Université de Versailles	**Rapporteur**
M. MICHEL KIEFFER, Hdr, Supélec	**Examinateur**
M. PIERRE SENS, Prof, Hdr, Université Pierre et Marie Curie	**Examinateur**
Mme. VERONIQUE VEQUE, Prof, Hdr, Université Paris-Sud	**Examinateur**
M. DJAMAL ZEGHLACHE, Prof, Hdr, Telecom SudParis	**Examinateur**

Thèse n° 2012TELE0021

APPROCHE CROSS-LAYER POUR SERVICES MULTIMEDIA ÉVOLUTIFS DISTRIBUÉS SUR LA PROCHAINE GÉNÉRATION DE RÉSEAUX CONVERGENTS : ARCHITECTURES ET PERFORMANCES

RESUME:

Multi-parti de conférence multimédia est le type le plus compliqué de la communication mais aussi le service principalement utilisé sur Internet. Il est aussi la killer application sur les réseaux 4G. Dans cette recherche, nous nous concentrons sur trois parties principales du service de téléconférence: L'architecture de distribution de médias, le codage vidéo, ainsi que l'intégration du service dans les infrastructures sans fil 4G. Nous proposons un algorithme d'application nouvelle couche de multidiffusion utilisant une architecture de services distribués. L'algorithme proposé estime que les limites de la perception humaine, tout en participant à une conférence vidéo afin de minimiser le trafic qui n'est pas nécessaire pour la session de communication. Riche des modèles théoriques de la perception basée proposé architecture distribuée, l'architecture traditionnelle centralisée et basée sur la perception architecture centralisée ont été construits en utilisant la théorie d'attente afin de refléter le trafic généré, transmises et traitées à l'pairs distribués, les dirigeants et le serveur centralisé. La performance des architectures a été pris en compte dans les différents aspects de la durée totale d'attente, le retard de point à point et le taux de service requis pour le débit total. Ces résultats aident le lecteur à avoir une vue globale des performances de la proposition dans une comparaison équitable avec les méthodes conventionnelles. Pour construire l'arbre de distribution des médias pour l'architecture distribuée, une fonction de coût nouvelle application-aware multi-variable est proposée. Il tient compte des besoins variables des applications et des mises à jour dynamiquement les ressources disponibles nécessaires pour parvenir à un nœud particulier sur l'arbre de distribution de l'ALM. Codage vidéo scalable est utilisé comme le principal multi-couche codec à la conférence. Afin d'évaluer la performance de la fonction de coût multi-variable nouvellement proposées dans une application dynamique et avancé environnement réseau sans fil, codage vidéo scalable (SVC) des transmissions sur un réseau overlay ALM construite sur un réseau sous-4G/WiMAX réelles ont été utilisées. Nous avons développé EvalSVC et l'utiliser comme plate-forme principale pour évaluer la fonction de coût proposé. Comme un problème commun, l'architecture distribuée nécessite que les pairs contribuent une partie de leur bande passante et capacité de calcul afin de maintenir la superposition mutuelle inter-connexion. Cette exigence se développe en un grave problème pour les utilisateurs mobiles et l'infrastructure sans fil, comme la ressource radio de ce réseau est extrêmement coûteux, et est l'une des raisons pour lesquelles l'architecture distribuée n'a pas été largement appliquée dans la prochaine génération (4G) des réseaux. C'est aussi la raison principale pour laquelle les services multimédias tels que vidéo-conférence doivent s'appuyer sur une architecture centralisée coûteuse construite sur un des contrôleurs des médias coûteux fonction des ressources (CRFM), via l'IMS (IP Multimedia Subsystem). Ce travail de recherche propose une nouvelle architecture distribuée utilisant la capacité de renseignement et extra, actuellement disponibles sur le LTE et stations de base WiMAX de réduire le besoin des débits que chacun a à fournir par les pairs afin de maintenir

le réseau overlay. Cette réduction permet d'économiser des ressources précieuses et de radio permet une architecture distribuée pour fournir des services de visioconférence sur les réseaux 4G, avec tous les avantages d'une architecture distribuée, comme la flexibilité, l'évolutivité, les petits retards et à moindre coût. De plus, cela peut être mis en œuvre avec une modification minimale de la plate-forme standardisée IMS et les infrastructures 4G, économisant ainsi les opérateurs et les fournisseurs de services d'investissements excessifs. Analyse théorique et prototypes ont également été fournis afin de prouver les avantages et la faisabilité des travaux de recherche.

Mots-clés : La livraison de contenu multimédia, des services multimédias, la couche d'application de multidiffusion, vidéo conférence, Scalable Video Coding, une évolutivité SNR, scalabilité spatiale, scalabilité temporelle, AVC, plate-forme d'évaluation vidéo, paramètres QoS et de mesure, de bout en bout la qualité de service, les temps d'attente, de bout en bout retard, analyse l'architecture, la plate-forme de simulation, l'évaluation des performances, des SVC outils d'évaluation, l'application couche de routage, la fonction de coût, la fourniture de ressources, optimisation inter-couches, goulot d'étranglement, couverture de réseau, les réseaux sans fil 4G, le WiMAX, les réseaux convergents IEEE802.16e, P2P, IMS, LTE, WiMAX, NGN, ALM, 4G, architecture de services, l'architecture distribuée, l'architecture centralisée, vidéo-conférence distribuée,

1.INTRODUCTION

Les services multimédias est la killer application sur les réseaux convergents de prochaine génération. Contenus vidéo sont la partie la plus consomme beaucoup de ressources d'un flux multimédia. La vidéo est prévu pour être le prochain communication populaire multimédia après les communications vocales. La transmission vidéo, multicast vidéo et services de vidéoconférence sont les types les plus populaires de la communication vidéo avec des niveaux de difficulté croissante. Quatre parties principales de la périphérie cross-layer évolutives des services multimédias sur des réseaux convergents de prochaine génération sont pris en compte dans ce travail de recherche, tant du point l'architecture et les performances des points de vue.

Les gens sont maintenant de travail et de divertissement dans un “3-écran” monde. Ces écrans sont différents dans leurs capacités de calcul, des résolutions d'écran, et des bandes passantes de communication. Une bien meilleure solution que AVC consiste à utiliser codage vidéo scalable (SVC). SVC a été normalisé comme une extension de la norme AVC depuis 2007. L'idée principale de cette extension consiste à appliquer plusieurs couches de codage dans le codec AVC. Ce n'est pas une idée totalement nouvelle puisque les gens ont tenté de mettre en œuvre cette idée de précédentes normes internationales de codage vidéo tels que H.262/MPEG.2 vidéo, H.263 et MPEG.4 visuelle. Cependant, le problème le plus difficile, c'est que, l'évolutivité habitude de venir avec une énorme augmentation de la complexité de calcul. SVC a réussi à fournir une évolutivité à un coût abordable de calcul. SVC code pour un flux vidéo d'entrée en une sortie multi-couche flux de bits comprenant une couche de base et plusieurs couches d'amélioration. Au sein de ces couches, la couche de base est codé avec une base de qualité pour garantir qu'il puisse être consommé par le plus faible du récepteur du groupe communication dans son ensemble. Cette couche de base est généralement protégé tout en étant transmis sur le réseau de transmission par des méthodes de qualité de service assuré ou Forward Error Correction (FEC) des algorithmes. Aux fins de compatibilité ascendante, la couche de base doit être reconnu par tous les décodeurs H.264 classiques. Couches d'amélioration, lorsque reçus au niveau des récepteurs en même temps que la couche de base, d'accroître l'ensemble de qualité de l'flux de bits. Particulier, lorsque tous les couches d'amélioration sont reçus dans l'ordre au-récepteur avec la couche de base, le train binaire atteindra sa qualité d'origine codé. Toutefois, lorsque les conditions réelles (comme des bandes passantes, des retards, ou des tailles d'écran d'affichage) ne permettent pas, les couches supérieures peuvent être jetés le long de la ligne de transmission ou à une boîte de milieu (entités de relais) pour le flux de bits pour être en forme-avec ces conditions sans pour autant corrompre la session de communication vidéo.

Services de vidéo à l'aide SVC ont été lancés depuis la normalisation du codec SVC. SVC codec est dédié à améliorer les performances de transmission et non l'exécution de codage du contenu vidéo. Cependant, peu de travaux ont été menées sur l'évaluation de bout en bout la transmission des performances des matières SVC. Afin d'évaluer la performance de transmission de bout en bout de contenus vidéo SVC, les concepteurs et les chercheurs sont vraiment dans le besoin d'un outil d'évaluation de transmission vidéo qui est spécialement adapté pour l'évaluation de la SVC transmissions vidéo codés sur un réseau réel ou simulé. Jusqu'à présent, la communauté de recherche dépend de Evalvid pour mesurer l'évaluation de la transmission du contenu AVC. Evalvid ne peut supporter le codec vidéo H.264. Il ne peut pas prendre la vidéo SVC comme une entrée possible au processus d'évaluation. En outre, Evalvid est limitée dans son interface d'Simulator seul vrai et du réseau (NS-2) environnement de réseau basée sur. Sorties Evalvid, limitée à seulement deux paramètres: Signal to Noise Ratio de crête (PSNR) et Mean Opinion Score (MOS), ne reflètent pas bien la

performance de la transmission vidéo. Dans notre plate-forme EvalSVC nouvelle, nous parvenons à surmonter toutes ces limitations en soutenant l'évaluation de la transmission SVC. En plus des mesures classiques de PSNR et MOS, nous utilisons aussi l'indice de similarité structurelle (SSIM). SSIM est un procédé de mesure de la similitude entre deux images. Il a été conçu pour améliorer les méthodes traditionnelles comme crête de signal-à-bruit (PSNR) et erreur quadratique moyenne (MSE), qui se sont révélés incompatibles avec la perception œil humain. Dans notre plate-forme EvalSVC nouvelle, plus les interfaces avec le goulot de la bouteille et des réseaux de recouvrement sont développés. Depuis les goulots d'étranglement peuvent être trouvées partout sur le réseau et il est effectivement un problème sérieux pour les services multimédias, l'interface avec le réseau goulot d'étranglement est vraiment nécessaire pour l'évaluation de transmission SVC. L'interface de la couche réseau est nécessaire parce que superposition de réseau est l'architecture de transport futur des services multimédias (tels que le peer-to-peer communication et basés sur le cloud services multimédias).

En comparaison avec la plate-forme d'évaluation classiques (tels que EvalVid), le principal avantage de l'EvalSVC nouvellement proposé est sa capacité à soutenir SVC-codées contenu. En outre, EvalSVC a enrichi des interfaces pour la plate-forme de simulation de recouvrement (construit par Oversim).
Dans cette thése , nous introduisons également des interfaces de la EvalSVC à de véritables réseaux tels qu'Internet réel ou réels des réseaux sans fil (GPRS, UMTS, WiMAX, LTE, WiFi), à un environnement réseau hybride, et à la plate-forme de simulation comme le recouvrement échantillons d'interfaces possibles à la plate-forme. Ces interfaces de réseaux réels et des réseaux hybrides ont été appliquées pour l'évaluation de la transmission vidéo SVC sur un réseau WiMAX mobile réel et un réseau hybride composée d'une topologie de l'Internet simulé et un réseau WiMAX mobile réel. Les interfaces réelles aider EvalSVC pour obtenir des mesures réelles de la session de transmission vidéo sur les réseaux réels. L'interface permet Oversim EvalSVC pour simuler un environnement réseau de recouvrement de telle sorte que distribuée SVC une prestation axée sur le contenu vidéo peut être évalué avec notre EvalSVC. Plus d'interfaces peuvent être ajoutés à cette plate-forme open source sur les besoins et les exigences des communautés de recherche et industriels.
Pour la fonction de coût sur lequel l'on peut construire, nous proposons une fonction d'application-aware coût multi-variable.
Multidiffusion est le procédé de délivrance de données sur un groupe de pré-enregistrés destinations. Conceptuellement, le multicast a une meilleure performance que les deux unicast et diffusé au moment de servir seulement un certain groupe d'utilisateurs. L'Internet a été construit pour unicast ou one-to-one applications. De nos jours, il doit servir un grand nombre de services multimédias telles que la conférence multimédia ou jeux multi-joueurs. Ces types de services de multidiffusion mis une grosse charge sur l'infrastructure unicast de l'Internet. Par conséquent, il existe une demande pour la conception et le déploiement d'algorithmes multicast sur Internet.
Multicast peut être approché à partir de la couche réseau soit ou de la couche application. En ce qui concerne la multidiffusion dans la couche réseau, l'IP-Multicast est la première tentative pour résoudre ce problème. Il est si loin le mécanisme le plus efficace de multidiffusion pour fournir des données sur chaque lien du réseau qu'une seule fois. Cependant, de nombreux problèmes sont encore le déploiement de la prévention de l'IP-Multicast d'être pris en charge dans le monde entier.
Des tentatives ont été faites pour surmonter ces problèmes. Explicit multi-unicast (Xcast) est une autre stratégie de multidiffusion IP multicast qui fournit les adresses de réception de toutes les destinations au sein de chaque paquet. En tant que tel, puisque la taille des paquets

IP est limitée, en général, Xcast ne peut pas être utilisé pour des groupes de multidiffusion de grand nombre de destinations.
Une solution alternative pour Multicast est de construire Multicast Application Level (ALM) sur le réseau sous-jacent. Le concept clé de l'ALM est la mise en œuvre de la multi-fonctionnalité de coulée comme un service d'application au lieu d'un service réseau. Il a d'excellents avantages sur IP-Multicast: déploiement facile et peut-être immédiate sur Internet sans aucune modification de l'infrastructure actuelle et adaptable à une application spécifique. Arbre-poussoir est une approche commune pour la livraison de données dans les algorithmes de la couche d'application de multidiffusion, en particulier lorsque la qualité multimédia est préoccupé. Dans cette approche, avant la distribution des données ne peut avoir lieu, un arbre de distribution des médias doit être construit à partir de tous les pairs participants, puis les données sont activement distribué à partir du noeud source pairs intermédiaires jusqu'à ce qu'il atteigne tous ses pairs dans l'arbre multicast. Afin de construire cet arbre de distribution ALM, nous devons avoir des coûts de tous les disponibles fin-en-bout des liens. Ces coûts ne peuvent être calculées en utilisant une fonction de coût. L'algorithme de construction de l'arbre de distribution de médias est fondée sur les coûts entre les pairs participants. Finalement, l'efficacité de l'algorithme de distribution des médias dépendra principalement de la fonction de coût utilisée.
La fonction de coût envisagée ne devrait pas être confondu avec des fonctions de réseau classiques des coûts d'après un calque, car ils ne fonctionnent pas sur la même couche. En fait, la fonction de coût de la couche applicative repose sur la fonction de coût de la couche réseau pour le routage sous-jacent et ne sera responsable de l'acheminement sur le niveau de la couverture de la multidiffusion de la couche application.

2.END-TO-END TRANSMISSION DES MATIÈRES VIDÉO SCALABLE: ÉVALUATION DE LA PERFORMANCE AU COURS EVALSVC-UNE PLATE-FORME NOUVELLE SOURCE OUVERT

Tout d'abord, nous évaluons la performance des transmissions multimédias évolutifs sur un réseau overlay. Pour cela, nous évaluons la performance de la vidéo évolutives de bout en bout des transmissions sur EvalSVC: une plate-forme nouvelle évaluation open-source. Scalable Video Coding (SVC) est la dernière extension de la vidéo Advance célèbre Coding (AVC) standard. L'évolutivité est important et utile, car elle est dédiée à la transmission de contenus vidéo sur les conditions de réseau hétérogènes et les capacités terminaux. Néanmoins, la communauté des services multimédias de recherche et de l'industrie n'ont pas été en mesure d'utiliser pleinement tout le potentiel de cette vidéo extension standard de codage en raison de l'absence d'une plate-forme d'évaluation pour évaluer la transmission de bout-en-fin de SVC-contenu. EvalSVC vise à favoriser SVC-applications et de recherche dans les services multimédia. Il est capable d'évaluer la transmission de bout-en-fin de SVC peu les flux encodés avec des fonctionnalités améliorées (spatiale, temporelle, SNR, et combiné évolutivité).

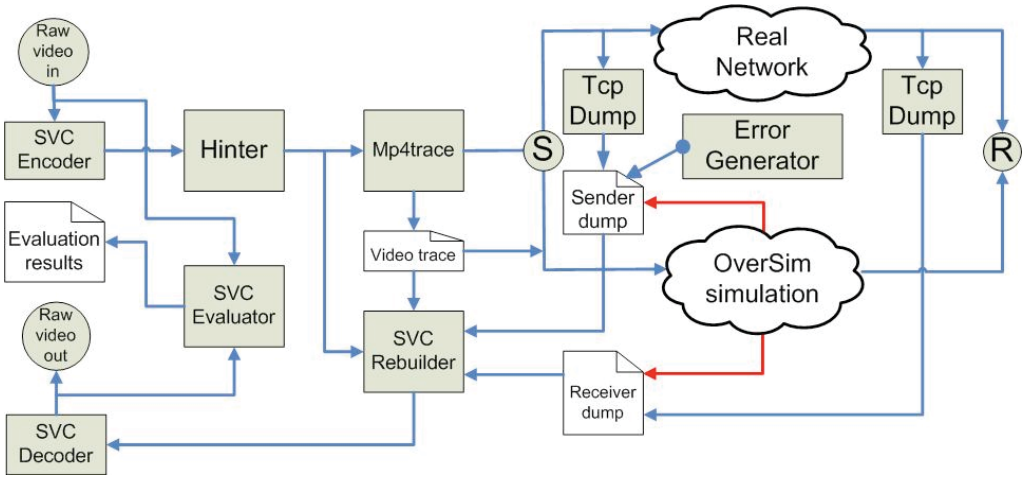


Diagramme d'EvalSVC.

Les résultats de sortie sont des mesures objectives et subjectives de la transmission vidéo. Interfaces avec les réseaux réels et une plate-forme de simulation de recouvrement sont présentés. Grâce à ces interfaces, les performances de transmission de différents types d'évolutivité SVC et AVC peu les flux sur un goulot d'étranglement et un réseau de recouvrement seront évaluées.

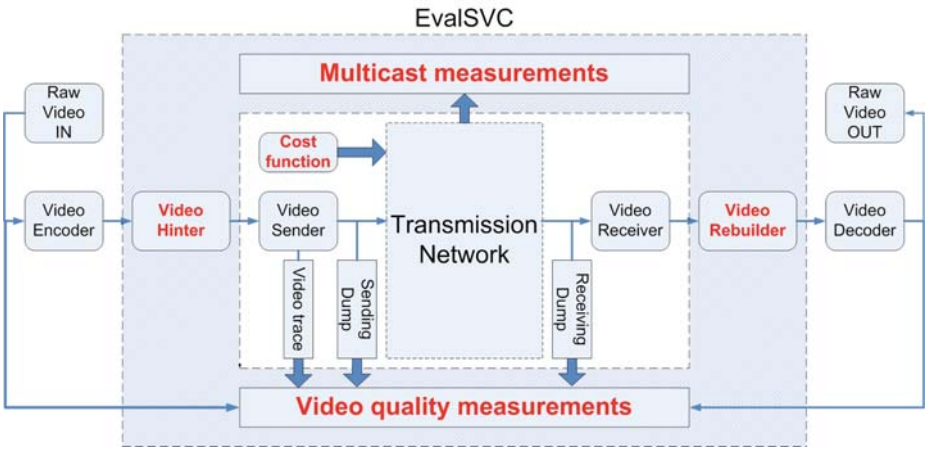
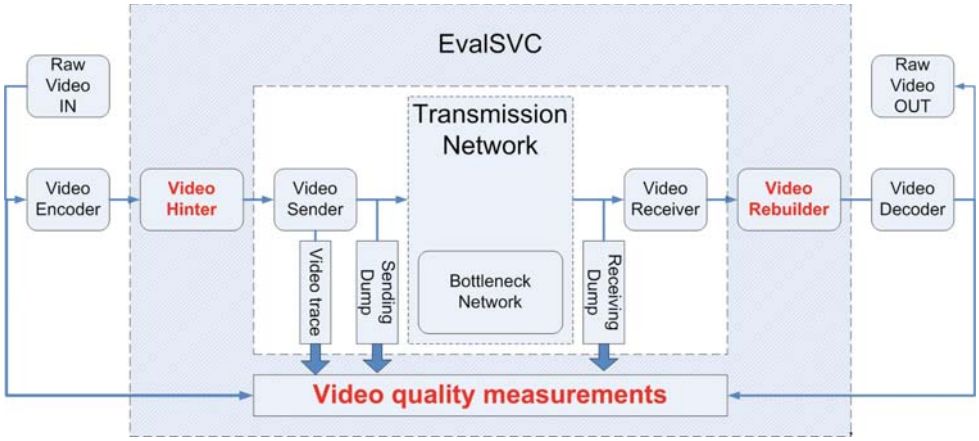


Schéma général.

Cette évaluation est nouvelle parce qu'elle est menée sur la transmission de bout en bout des matières SVC et non sur la performance de codage. L'environnement réseau goulot d'étranglement est mis en place pour l'évaluation parce que cette condition réaliste d'un réseau et très commune provoque généralement des problèmes graves pour la transmission vidéo et des services multimédias. Le réseau de recouvrement est utilisé pour l'évaluation, car il est un environnement de transport pour la livraison future de contenu multimédia. Les résultats montrent que, à la fois dans des conditions de réseau goulet d'étranglement et sur l'environnement réseau de recouvrement, la transmission SVC surpasse AVC de transmission et l'évolutivité SNR a la plus haute performance de transmission. Grâce à ces évaluations fin-en-bout des performances de transmission plus EvalSVC, nous pouvons conclure que, en effet SVC atteint ses objectifs d'amélioration de la transmission vidéo sur les conditions de réseau réalistes. Par conséquent, l'évolutivité SVC et plus spécifiquement SVC SNR sera utilisé dans les parties suivantes de la recherche que le schéma de codage vidéo principale. Après avoir

trouvé l'évolutivité adaptée SVC pour un service multimédia, nous allons étudier le mécanisme de multidiffusion pour le contenu multimédia sur un réseau overlay dans la partie suivante de cette thèse.



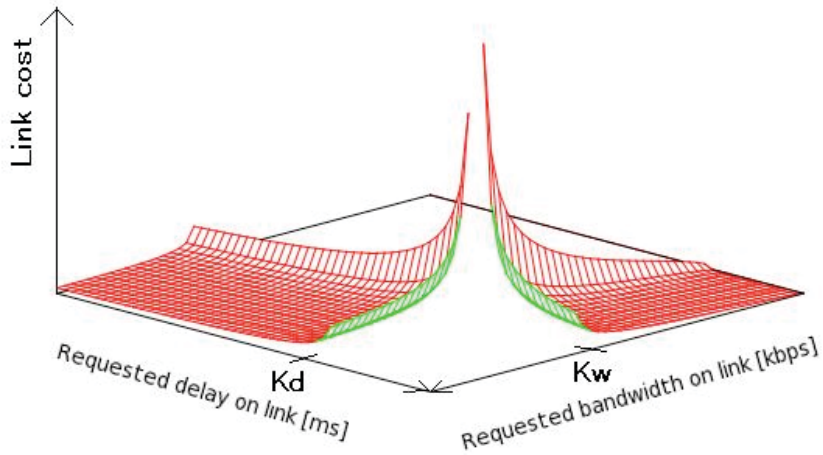
Goulot de la bouteille diagramme.

Cette évaluation est nouvelle parce qu'elle est menée sur la transmission de bout en bout des matières SVC et non sur la performance de codage. L'environnement réseau goulot d'étranglement est mis en place pour l'évaluation parce que cette condition réaliste d'un réseau et très commune provoque généralement des problèmes graves pour la transmission vidéo et des services multimédias. Le réseau de recouvrement est utilisé pour l'évaluation, car il est un environnement de transport pour la livraison future de contenu multimédia. Les résultats montrent que, à la fois dans des conditions de réseau goulet d'étranglement et sur l'environnement réseau de recouvrement, la transmission SVC surpasse AVC de transmission et l'évolutivité SNR a la plus haute performance de transmission. Grâce à ces évaluations fin-en-bout des performances de transmission plus EvalSVC, nous pouvons conclure que, en effet SVC atteint ses objectifs d'amélioration de la transmission vidéo sur les conditions de réseau réalistes. Par conséquent, l'évolutivité SVC et plus spécifiquement SVC SNR sera utilisé dans les parties suivantes de la recherche que le schéma de codage vidéo principale. Après avoir trouvé l'évolutivité adaptée SVC pour un service multimédia, nous allons étudier le mécanisme de multidiffusion pour le contenu multimédia sur un réseau overlay dans la partie suivante de cette thèse.

3. DEMANDE-LA COUCHE RÉSEAU DE LA CROIX COÛT FONCTION MULTI-VARIABLE POUR LA LIVRAISON MULTIMÉDIA SUR DES RÉSEAUX CONVERGENTS

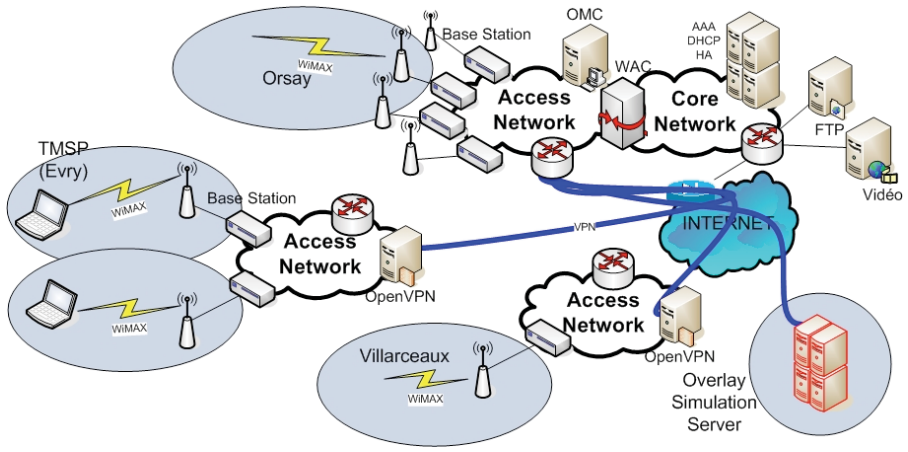
Deuxièmement, nous nous attaquons à des problèmes de la répartie cross-layer multicast multimédia évolutive sur le réseau de prochaine génération convergente (la convergence entre un réseau sans fil 4G et un réseau fixe). Pour cela, nous proposons une nouvelle application de la couche réseau croix fonction de coût multi-variable pour la multidiffusion de couche d'application de la livraison multimédia sur des réseaux convergents. Application Layer Multicast (ALM) algorithmes sont similaires ou conceptuellement basée sur les fonctions de coût de la couche réseau de multidiffusion de. Une nouvelle application de la couche réseau croix fonction de coût multi-variable est proposé. Il optimise les exigences variables et les ressources disponibles de l'application et les couches du réseau. Il peut mettre à jour

dynamiquement les ressources disponibles nécessaires pour parvenir à un noeud particulier sur l'arbre de la gestion actif-passif de distribution des médias.



Fonction de coût à plusieurs variables

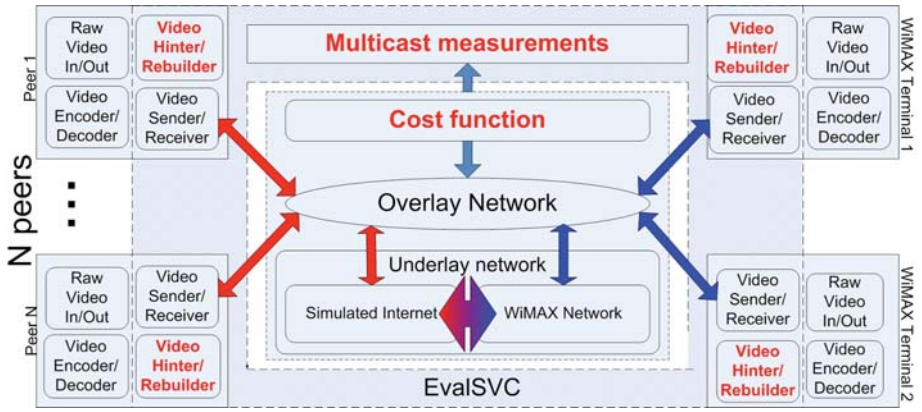
Dérivation mathématique et l'analyse théorique ont été fournies pour la fonction de coût nouvellement proposé afin qu'il puisse être appliqué dans des cas plus généraux de contextes différents. Une plate-forme d'évaluation d'un réseau overlay construit sur un réseau sous-jacent convergent composé d'une topologie de l'Internet simulé et une réelle 4G WiMAX mobile IEEE802.16e réseau sans fil est construit.



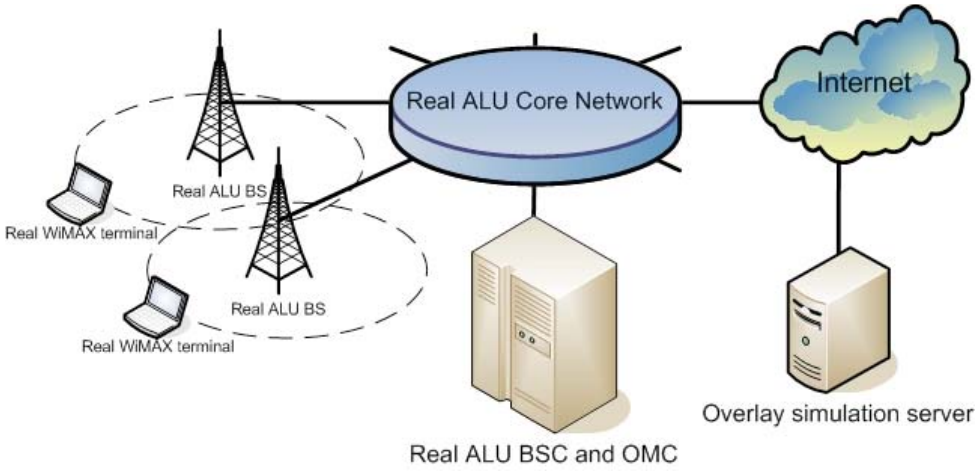
Réel réseau WiMAX

La performance de la nouvelle proposition cross-layer fonction de coût multi-variable est évaluée et comparée avec les fonctions de coût classiques dans Scalable Video Coding basés sur des services tels que la multidiffusion vidéo et des services de visioconférence en utilisant la plate-forme d'évaluation.

Résultats de l'évaluation intensifs ont montré que la proposition de nouvel-couche fonction de coût multi-variable surpasse la fonction de coût classique et améliore la performance des services multimédias sur réseau convergent environnements dynamiques. Si la multidiffusion est le seul mécanisme à plusieurs de distribuer le contenu multimédia, une étude plus approfondie sur le mécanisme de plusieurs-vers-plusieurs se fera dans la prochaine partie de la thèse par le biais d'une nouvelle architecture pour les services de vidéoconférence.



Interface avec le réseau WiMAX réel



Scénario d'évaluation

4.ENRICHI HUMAINE BASÉE SUR LA PERCEPTION ARCHITECTURE DISTRIBUÉE POUR ÉVOLUTIVES DES SERVICES DE VIDÉOCONFÉRENCE: MODÈLES THÉORIQUES ET DE LA PERFORMANCE

Troisièmement, nous étudions les distribués cross-layer évolutives des services de vidéoconférence sur le réseau overlay. Pour cela, un homme enrichi la perception basée sur une architecture distribuée pour la évolutives des services de vidéoconférence est proposé avec les modèles théoriques et analyse de la performance. La nouvelle architecture proposée peut effectivement réduire le trafic inutile sur les flux vidéo multi-couches transmises sur le réseau overlay.

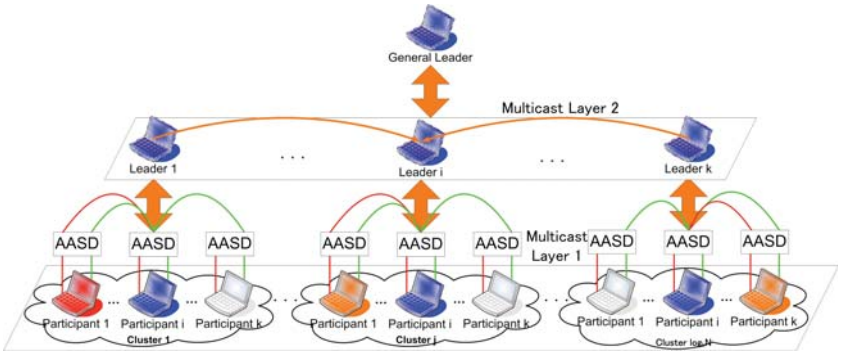
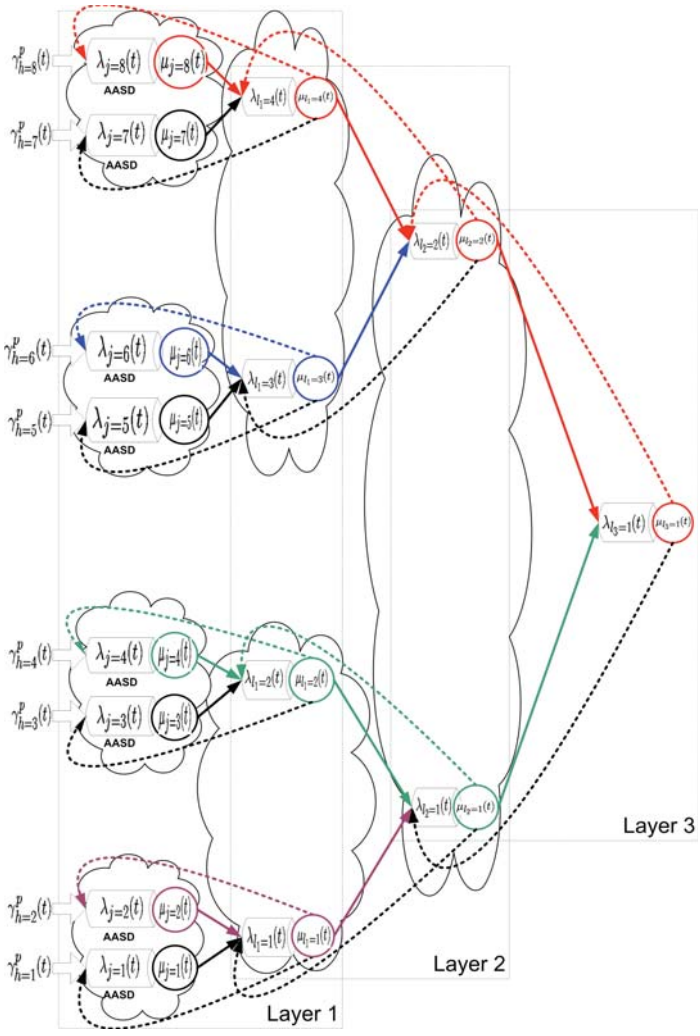
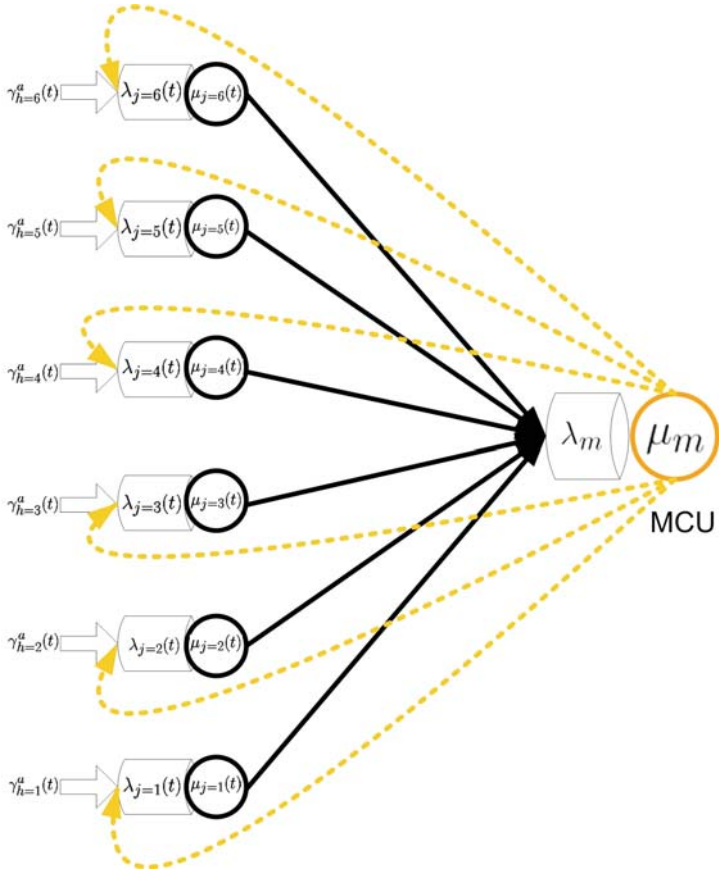


Diagramme d'architecture

Rich modèles théoriques des trois architectures différentes: le projet de la perception basée sur l'architecture distribuée, l'architecture classique et de la perception centralisée basée sur une architecture centralisée ont été construits en utilisant la théorie de files d'attente afin de refléter le trafic généré, transmises et traitées aux dirigeants perception basés distribués, la perception à base centralisée dirigeant dessus, et le serveur centralisé. La performance de ces trois architectures différentes a été pris en compte dans les différents aspects de la durée totale d'attente, le retard de point à point et les taux de service requis pour le débit total.



ALM modèle



MCU modèle

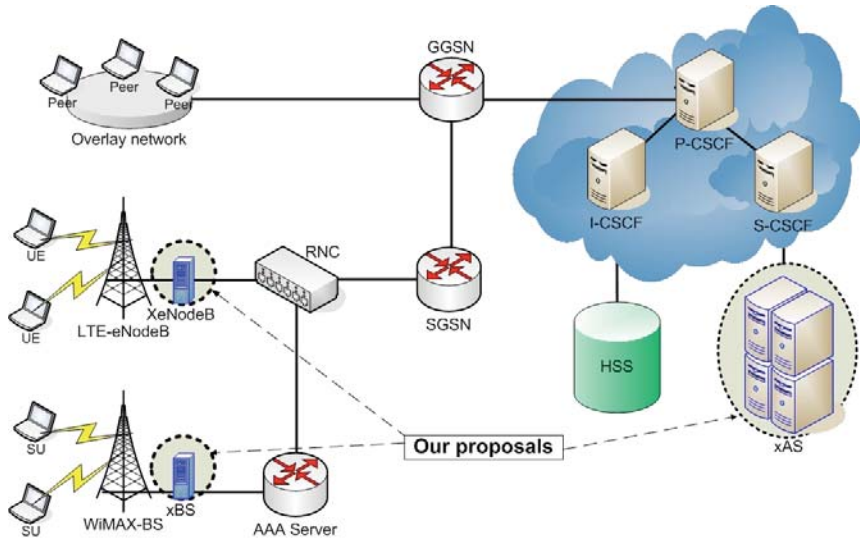
Les modèles théoriques et des analyses donnent un résultat de comparaison de meilleurs parmi les trois architectures différentes. Ces résultats aident les lecteurs à avoir une vision globale de la performance de la proposition dans une comparaison équitable avec les méthodes conventionnelles. Ensemble, les outils de modélisation, l'analyse et les résultats numériques aider à répondre à la préoccupation commune au sujet des avantages et des inconvénients entre les architectures centralisées et distribuées pour l'architecture du service de vidéoconférence.

L'ensemble, l'impression proposée humaine basé sur l'architecture distribuée pour pluripartites services de visioconférence peut maintenir un temps d'attente plus faible totale et le point à point avec un retard plus petite exigence de débit de service et un débit total d'équivalent par rapport à l'centralisée classique l'architecture et de la perception basée sur une architecture centralisée. Bien que l'architecture distribuée est meilleure que l'architecture centralisée pour un service de téléconférence multimédia évolutive, il apporte de nombreux problèmes aux utilisateurs qui utilisent un réseau sans fil à participer dans le service de conférence. Une solution particulière devrait être savoir pour les utilisateurs mobiles dans la prochaine partie de la thèse.

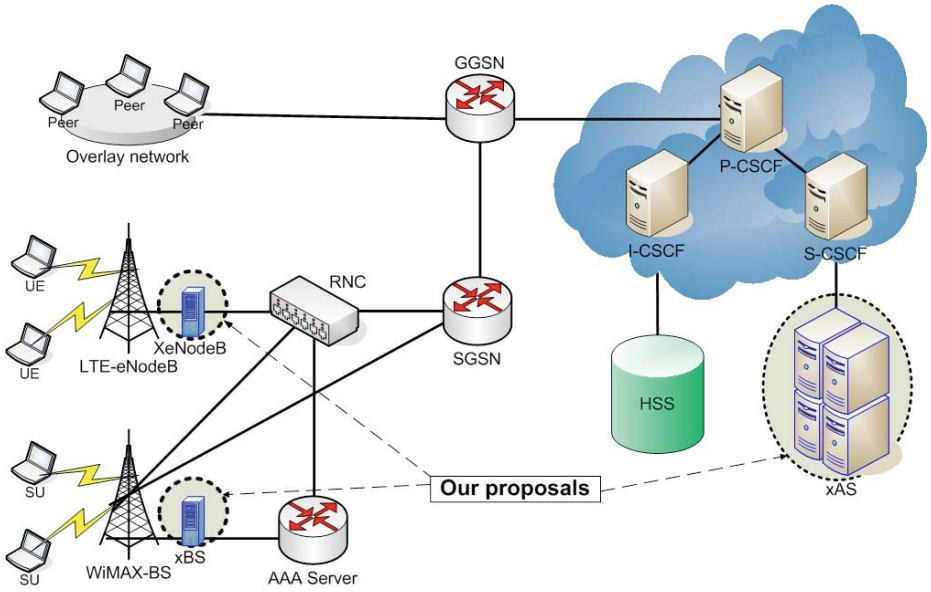
5.IMS BASÉE DISTRIBUÉS SERVICES DE CONFÉRENCE MULTIMÉDIA POUR LA PROCHAINE GÉNÉRATION DES RÉSEAUX MOBILES

“Last but not least”, les réparties cross-layer évolutives des services de vidéoconférence sur le réseau de prochaine génération convergente est activée. Pour cela, un IMS basée sur des

services de téléconférence multimédia distribués pour Next Generation Networks convergentes est proposé.

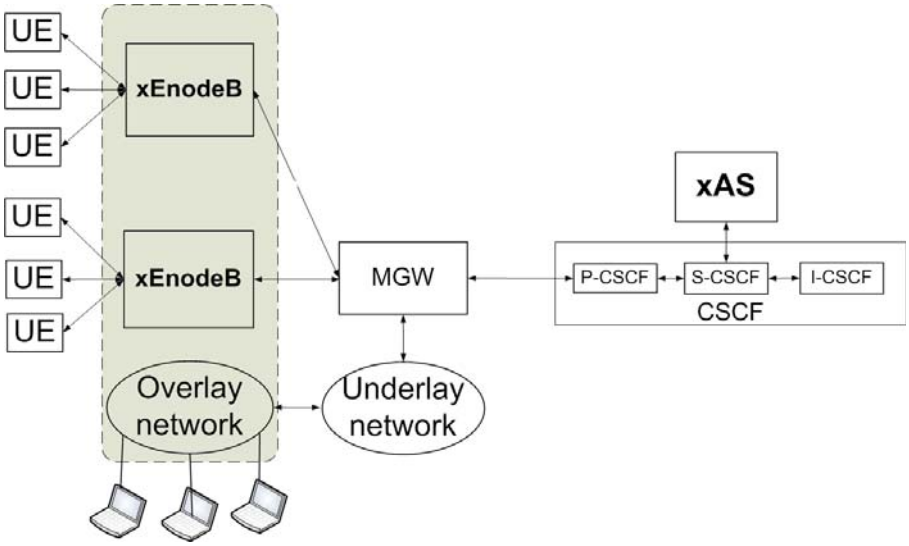


Couplage lâche



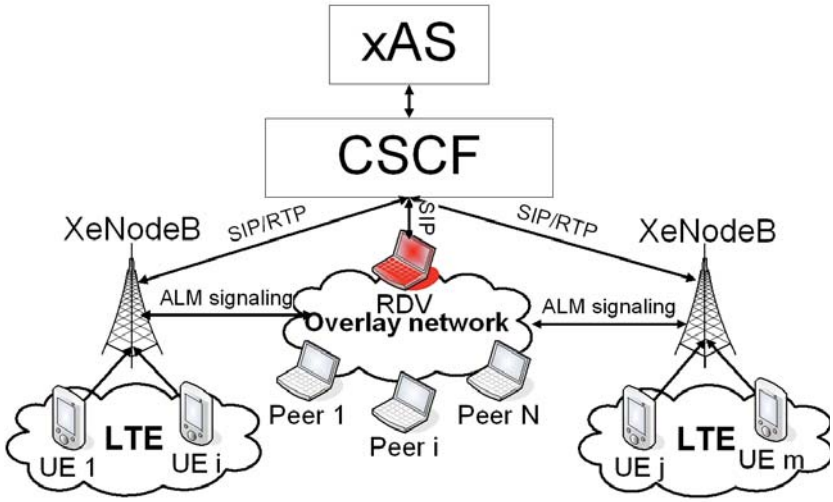
Couplage étroit

L'architecture distribuée offre de nombreux avantages par rapport à l'architecture centralisée en termes de fourniture de services multimédias. Cependant, comme un compromis, l'architecture distribuée exige que les pairs contribuent une partie de leur bande passante et capacité de calcul de maintenir la superposition mutuelle inter-connexion.



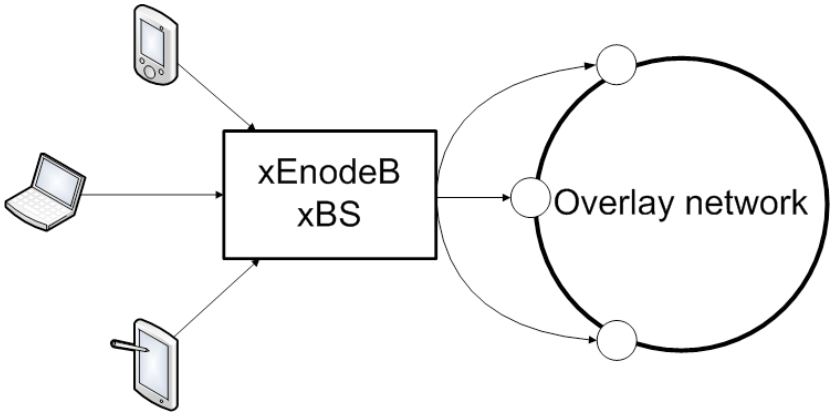
Vidéo-conférence Distribué

Cette exigence se développe en un grave problème pour les utilisateurs mobiles et l'infrastructure sans fil, car la ressource radio dans ce réseau est extrêmement coûteux, et est l'une des raisons pourquoi l'architecture distribuée n'a pas été largement appliqués dans la prochaine génération (4G) des réseaux. Il est également la principale raison pourquoi les services multimédias tels que vidéo-conférence doivent compter sur une architecture centralisée coûteux construit sur un des contrôleurs des médias coûteux fonction des ressources (CRFM) via l'IMS (IP Multimedia Subsystem).



LTE vidéoconférence Distribué

Cette partie du travail de recherche propose une nouvelle architecture distribuée utilisant l'intelligence et la capacité supplémentaire, actuellement disponible sur le LTE et stations de base WiMAX visant à réduire les débits nécessaires à ce que chaque poste est tenu de fournir en vue de maintenir le réseau de recouvrement. Cette réduction permet d'économiser de précieuses ressources radio et permet une architecture distribuée pour fournir des services de visioconférence sur les réseaux 4G, avec tous les avantages d'une architecture répartie, telle que la flexibilité, l'évolutivité, délai plus court et un coût moindre.



Noeuds représentatifs

En outre, cela peut être mis en œuvre avec une modification minimale de la plate-forme standardisée IMS et de l'infrastructure 4G, économisant ainsi les opérateurs et les fournisseurs de services provenant des investissements excessifs. Un prototype a été construit pour prouver la faisabilité de l'architecture proposée et d'évaluer ses performances. Notre projet de service distribué de visioconférence peuvent effectivement réduire la bande passante moyenne requise pour les données et les messages de signalisation à des terminaux mobiles sans fil tout en conservant les principales opérations d'une session de vidéo conférence.

6. CONCLUSION

En conclusion, à travers l'étude dans cette thèse, nous avons réussi à résoudre les problèmes d'évolutivité qui sont présentes sur les multi-partis, des services de conférence multimédia.
Comme nous comptons sur le Scalable Video Coding que le codec vidéo principale pour notre réseau de recouvrement évolutive basée sur l'architecture vidéo distribuée conférence pour résoudre le terminal évolutivité, nous avons réussi à construire une plate-forme d'évaluation (EvalSVC) pour évaluer évolutive des performances de transmission vidéo dans un environnement réseau divers tels que goulot d'étranglement, de superposition, et surtout pour comparer entre les plates-formes distribuées un centralisées de services de conférence. Sur la base des résultats obtenus à partir du processus d'évaluation, il est clair que l'architecture distribuée fondée sur Multicast Application Layer présente de nombreux avantages de l'architecture classique centralisée construit sur l'unité de contrôle multipoint surtout quand une vidéo multi-couche de codage tels que codage vidéo scalable est appliquée . Obtenir ce fait à partir des résultats de l'évaluation, la thèse a proposé une nouvelle fonction de coût multi-variable basé sur les exigences de l'application. Le processus de dérivation mathématique a également été décrit dans les détails afin que l'on peut l'appliquer pour obtenir d'autres fonctions de coût multi-variables en fonction de leurs besoins spécifiques. L'idée principale est de soutenir la couche application des algorithmes de routage avec les exigences de couche d'application et l'approche. La fonction de coût nouvellement proposé a examiné les exigences dynamiques de l'application et le réseau sous-jacent et fait une optimisation inter-couches pour la fonction de coût multi-variable nouvellement proposé. L'analyse théorique a montré que la proposition de nouvel-couche multi-fonction de coût variable peut tenir compte simultanément des diverses exigences de l'application ainsi que la ressource possible à partir du réseau. Résultats de l'évaluation intensives avec réseau d'accès réel WiMAX ont montré que, dans un état réel sans fil, et avec ALM des services de multidiffusion, la fonction de coût récemment proposé à plusieurs variables peuvent encore

mieux s'adapter à l'évolution rapide des ressources disponibles ainsi que l'application dynamique de exigences que les fonctions de coût classiques. Cependant, si une fonction de coût multi-variable peut tenir compte de nombreux paramètres de qualité de service dans le même temps, il convient de noter que la fonction de coût multi-variable ne donne pas toujours un meilleur résultat que la fonction de coût variable unique dans certains cas particuliers.
La plate-forme nouvellement proposé EvalSVC et la fonction de coût multi-variable sont une combinaison de solution innovante pour la création et l'évaluation des futurs services multimédias basés sur Multicast couche d'application et Scalable Video Coding pour les nouvelles technologies d'accès sans fil comme le WiMAX.
Afin de résoudre l'évolutivité architecture, une nouvelle architecture distribuée enrichi de vidéoconférence compte tenu de la limitation de la perception humaine a été proposé. Modèles d'analyse mathématiques ont été construites et comparées pour centralisée, basée sur la perception et la perception centralisée basée sur des architectures distribuées en utilisant la théorie de files d'attente en termes de temps total, en attendant point-à-point de retard, les taux de service requis, et le débit total. Il est intéressant de remarquer que toutes les fonctionnalités enrichies de l'architecture de visioconférence proposée ont été modélisés en détail et inclus dans l'analyse mathématique et des expressions. Les simulations numériques obtenues à partir des modèles d'analyse théoriques et les données off-line statistiques ont été réalisées dans le cadre d'un multi-partie multi-couches service de vidéoconférence. Les résultats des simulations montrent que la nouvelle proposition basée sur la perception architecture distribuée peut effectivement réduire le temps d'attente total, le délai de bout en bout, le taux de service requis en comparaison avec les architectures centralisées et la perception basée sur centralisés. En ce qui concerne le débit total, la perception de l'architecture distribuée basée sur avoir un trafic équivalent avec les architectures centralisées et la perception basée sur centralisés. Lorsque le mécanisme de perception basé sur est appliquée, le débit total de la proposition est encore plus faible que les deux architectures classiques. Le résultat de l'analyse théorique a montré un grand avantage de la perception basée sur l'architecture distribuée contre l'architecture centralisée et la perception de l'architecture basée sur centralisée en particulier lorsque le nombre total de participants augmente de conférence.
Enfin, pour résoudre la connectivité évolutive entre l'infrastructure ligne filaire (représenté par les participants basés sur Internet) et le sans fil basée sur l'infrastructure mobile (représenté par les avancées 4G WiMAX / LTE participants base), nous propose une nouvelle architecture pour l'inter-connectivité entre les UEs en cours d'exécution sur l'infrastructure LTE / WiMAX et en participant à une conférence de recouvrement à base distribuée. Les résultats expérimentaux à partir du prototype ont montré une forte réduction du trafic de signalisation ainsi que dans le trafic de données gérée par chaque UE et par le cœur de réseau. Le débit moyen nécessaire à des terminaux sans fil dans notre architecture distribuée est équivalente à celle de l'architecture centralisée alors qu'elle est beaucoup inférieure à celle requise pour les réseaux de recouvrement classiques.
Scénarios de conférence divers tels que joindre / quitter, mettre en pause / retour, la couche d'application de transfert, et battements de cœur ont été pris en compte dans le prototype. La contribution principale est qu'il permet une conférence superposition vidéo basé sur avec tous les avantages d'une architecture distribuée et sans les inconvénients de trop de trafic en soulignant les connexions des terminaux mobiles sans fil (un critère obligatoire pour les réseaux sans fil). Le débit nécessaire de pairs sans fil dans notre architecture distribuée proposée est équivalente à celle d'un cas standard centralisé (basé sur les résultats obtenus à partir de l'évaluation de notre prototype de services audio et vidéo conférence distribués). Les résultats confirment qu'il est possible d'appliquer des architectures distribuées dans la prochaine génération de réseaux sans fil. Profitant de stations de base 4G signifie que le coût total des services de conférence peut être réduite et que ces services peuvent être fournis avec

un minimum de modifications des normes pertinentes. La proposition remplace l'architecture standard centralisé de la conférence IMS basée sur une solution plus robuste utilisant l'intelligence et la capacité de calcul de BS 4G (s). Le prototype de notre architecture distribuée proposée pour un service multimédia a montré qu'il peut être bien intégrés dans un réseau 4G. Sa nature distribuée conduit à une réduction considérable des coûts ainsi que plus de flexibilité et d'évolutivité combiné avec délai plus court. L'architecture peut aussi être appliqué dans les réseaux WiMAX avec une quantité raisonnable de modifications. La sécurité de cette architecture distribuée peut être hérité des mécanismes d'authentification et de chiffrement appliquées par LTE, WiMAX et les technologies SIP.

Nos bénéfices solution de tous les avantages d'une architecture distribuée, comme la flexibilité, une meilleure évolutivité, délai plus court et un coût moindre. Comme un compromis, une partie de la capacité de calcul de l'infrastructure (cœur de réseau et eNodeB / XBS) est nécessaire pour permettre à l'architecture distribuée dans un réseau mobile. Toutefois, cette situation n'est pas vraiment un inconvénient, car la capacité de calcul nécessaire à la contrepartie le serveur centralisé des médias (CRFM) est maintenant répartie sur l'infrastructure du réseau (eNodeB (s) / XBS (s) et le réseau de base).

Le seul inconvénient de cette solution sont les exigences de calcul / l'intelligence exigées des stations de base. En 2G/3G, cette solution pourrait ne pas être possible parce que le BS étaient de simples antennes avec très peu de capacité et de l'intelligence et les contrôleurs de station de base (BSC) sont séparés de stations de base. En 4G, le BSC et le BSS sont intégrés en une seule entité les XBS (dans le WiMAX) ou le (eNodeB LTE en). Par conséquent, l'intelligence et la capacité du BSC sont maintenant disponibles dans XBS / eNodeB. Ils peuvent désormais en charge une architecture distribuée et de faire notre proposition d'un objectif réalisable.

En résumé, pour les services multimédias construits sur un réseau convergent avec différents types de terminaux des utilisateurs, nous avons constaté que, l'architecture distribuée est meilleure que l'architecture centralisée spécifiquement en termes de temps d'attente total, de bout en bout, les taux de retard de service requis, et le débit total. Une optimisation cross-layer peut offrir de nombreux avantages car elle permet de modéliser et d'optimiser plus précisément deux de l'application des exigences et des ressources disponibles du réseau. Cette optimisation inter-couches peuvent réduire le trafic inutile transmis sur le réseau au cours d'une session de communication. Lorsqu'elle est appliquée dans une fonction de coût multi-variable, cette optimisation cross-layer peut construire un réseau de recouvrement optimisé pour la livraison du contenu multimédia. L'évolutivité est une caractéristique essentielle pour les services multimédias à surmonter les problèmes de congestion communes du réseau de transport. Nous avons également proposé une solution pour les services multimédias construits sur un réseau convergent pris en compte l'ensemble de l'architecture, distribuée optimisation inter-couches et d'évolutivité. Nous avons également prouvé que c'est une solution possible qui peut être intégré dans la plus à jour l'infrastructure de réseau convergente.

Acknowledgments

Though I am happy for having reached this point of my PhD student life, I cannot forget to save my gratitudes to the people who have been helping me to archive this milestone.

I would like to give many thanks to Prof. **Hang Nguyen** for sharing her scientific view of the research work, for her meticulous corrections and guidances. It is great that we can together solve the problems along the way to the common targets. Thank you for all of your supports.

It is my honour to have the direction from Prof. **Noel Crespi** without this, my PhD thesis cannot be completed successfully. The acceptances of all the committee members are really encouraging to me. They help me to finish the last difficult step of this PhD thesis with a great pleasure. Thank you very much for your time and your support.

My thanks to M. **Quang Hoang Nguyen**, M. **Quoc Tuan Tran** for their contribution in the development of the prototyping systems used in this thesis. Many thanks to Dr. **Hongguang Zhang** for his useful discussions, M. **Franck Gillet** and M. **Robin Carnino** for their technical supports, and the PhD students and the administrative staff in the team for their experience sharing, their collaborations and supports.

I am lucky to have the chance to live, play and work with two friendly Vietnamese communities: (i) in Evry (the "**Tapis group**") and (ii) in Paris ("**VNTelecom**"). Thanks for all of the time and experience we shared together and good luck with your careers and your lives.

It is really amazing when you do your PhD thesis abroad with your family members living with you. I highly recommend that. My love stays with my lovely wife, **Ngoc**, and my two little sons, **Ech** and **Bill**, for sharing with me all the ups and downs on the three-and-a-half-year long road. We've made it together!

*"Con cam on **Bo-Ma** vi da luon ung ho va dong vien nhung lua chon cua con. Cam on chu **Thanh** vi da giup do anh cham soc ong ba bo me khi anh dang mai me chu du thien ha de hoc nhung bai hoc ve su tu te".*

"In the end, only kindness matters!" - Hands lyrics - JEWEL.

Abstract

Multimedia services are the killer applications on next generation convergent networks. Video contents are the most resource consuming part of a multimedia flux. Video transmission, video multicast and video conferencing services are the most popular types of video communication with increasing difficulty levels. Four main parts of the distributed cross-layer scalable multimedia services over next generation convergent networks are considered in this research work, both from the architecture and performance point of views. Firstly, we evaluate the performance of scalable multimedia transmissions over an overlay network. For that, we evaluate the performance of scalable video end-to-end transmissions over EvalSVC. It is capable of evaluating the end-to-end transmission of SVC bit-streams. The output results are both objective and subjective metrics of the video transmission. Through the interfaces with real networks and an overlay simulation platform, the transmission performance of different types of SVC scalability and AVC bit-streams on a bottle-neck and an overlay network will be evaluated. This evaluation is new because it is conducted on the end-to-end transmission of SVC contents and not on the coding performance. Next, we will study the multicast mechanism for multimedia content over an overlay network in the following part of this PhD thesis. Secondly, we tackle the problems of the distributed cross-layer scalable multimedia multicast over the next generation convergent networks. For that, we propose a new application-network cross layer multi-variable cost function for application layer multicast of multimedia delivery over convergent networks. It optimizes the variable requirements and available resources from both the application and the network layers. It can dynamically update the available resources required for reaching a particular node on the ALM's media distribution tree. Mathematical derivation and theoretical analysis have been provided for the newly proposed cost function so that it can be applied in more general cases of different contexts. An evaluation platform of an overlay network built over a convergent underlay network comprised of a simulated Internet topology and a real 4G mobile WiMAX IEEE802.16e wireless network is constructed. If multicast is the one-to-many mechanism to distribute the multimedia content, a deeper study on the many-to-many mechanism will be done in the next part of the thesis through a new architecture for video conferencing services. Thirdly, we study the distributed cross-layer scalable video conferencing services over the overlay network. For that, an enriched human perception-based distributed architecture for scalable video conferencing services is proposed with theoretical models and performance analysis. Rich theoretical models of the three different architectures: the proposed perception-based distributed architecture, the conventional centralized architecture and perception-based centralized architecture have been constructed by using queuing theory to reflect the traffic generated, transmitted and processed at the perception-based distributed leaders, the perception-based centralized top leader, and the centralized server. The performance of these three different architectures has been considered in 4 different aspects. While the distributed architecture is better

than the centralized architecture for a scalable multimedia conferencing service, it brings many problems to users who are using a wireless network to participate into the conferencing service. A special solution should be found out for mobile users in the next part of the thesis. Lastly, the distributed cross-layer scalable video conferencing services over the next generation convergent network is enabled. For that, an IMS-based distributed multimedia conferencing services for Next Generation Convergent Networks is proposed. Distributed architecture offers many advantages compared to centralized architecture in terms of providing multimedia services. However, as a trade-off, distributed architecture requires that peers contribute a portion of their bandwidth and computational capacity to maintain the mutual overlay inter-connection. This requirement develops into a serious problem for mobile users and wireless infrastructure, as the radio resource in this network is tremendously expensive. It is also the main reason why multimedia services such as video conference have to rely on a costly centralized architecture built over an expensive Media Resource Function Controllers (MRFC) via the IMS (IP Multimedia Subsystem). This part of the research work proposes a new distributed architecture utilizing intelligence and extra capacity, currently available on LTE and WiMAX's Base Stations to reduce the required bit-rates that each peer has to provide in order to maintain the overlay network. This reduction saves valuable radio resources and allows a distributed architecture to provide video conferencing services on 4G networks, with all the advantages of a distributed architecture such as flexibility, scalability, smaller delay and lower cost. A prototype has been built to prove the feasibility of the proposed architecture and evaluate its performances. In conclusion, for multimedia services built over a convergent network with different types of users' terminals, we found that, distributed architecture is better than the centralized architecture specifically in terms of total waiting time, end-to-end delay, required service rates, and total throughput. A cross-layer optimization can offer many advantages because it can model and optimize more accurately both the application's requirements and the network's available resource. This cross-layer optimization can reduce the unnecessary traffic transmitted over the network during a communication session. When applied in a multi-variable cost function, this cross-layer optimization can build an overlay network optimized for multimedia contents delivery. Scalability is an essential feature for the multimedia services to overcome the common congestion problems of the transmission network. We have also proposed a solution for the multimedia services built over a convergent network taken into account all of the distributed architecture, cross-layer optimization and scalability. We have also proved that it is a possible solution which can be integrated into the most up-to-date convergent network infrastructure.

Keywords: Application layer multicast, video conference, Scalable video coding, video evaluation platform, architecture analysis, application layer routing, cost function, cross-layer optimization, overlay network, convergent networks, distributed architecture, centralized architecture,

Contents

1 Introduction **3**

2 End-to-End Transmission of Scalable Video Contents: Performance Evaluation over EvalSVC-A New Open Source Platform **19**

2.1 Introduction 19
2.2 Proposed EvalSVC Platform Description 20
2.2.1 Architecture description 20
2.2.2 Functional blocks 21
2.2.3 Evaluation metrics 23
2.3 Proposed EvalSVC's interfaces 24
2.3.1 Proposed interface to real networks 24
2.3.2 Proposed interface to Overlay simulation platform 25
2.3.3 Hybrid network environment 25
2.4 Performance comparison of AVC vs. different types of SVC end-to-end transmission on EvalSVC in a bottleneck network environment 26
2.5 Performance comparison of AVC vs. different types of SVC end-to-end transmission on EvalSVC over an overlay network environment 29
2.6 Conclusion 30

3 Application-Network Cross Layer Multi-variable Cost Function for multimedia delivery over Convergent Networks **33**

3.1 Introduction 33
3.2 Conventional approaches 34
3.3 Proposed cross-layer multi-variable cost function 36
3.3.1 Problem formulation 36
3.3.2 Lemma 1: Bandwidth-type cost function 36
3.3.3 Lemma 2: Delay-type cost function 38
3.3.4 Derivation of the multi-variable cost function 38
3.3.5 N-variable cost function with $N \geq 3$ 42
3.4 Theoretical analysis of the newly proposed cross-layer multi-variable cost function 42
3.5 Proposed cross-layer cost function performance on multimedia delivery platforms 43
3.5.1 Video multicast service on overlay network 45
3.5.2 Video conferencing service in convergent underlay network environment 48
3.6 Conclusion 51

4 Enriched human perception-based distributed architecture for scalable video conferencing services: theoretical models and performance 53
4.1 Introduction . . . 53
4.2 Proposed Architecture of Human Perception-based Distributed Scalable Video Conferencing Services . . . 54
4.3 Theoretical analysis . . . 56
4.3.1 Theoretical expression of the waiting time in queues . . . 61
4.3.2 Numerical calculation of the waiting time . . . 68
4.3.3 Point-to-point delay . . . 72
4.3.4 Required service rate . . . 76
4.3.5 Estimation of global throughput in the network . . . 78
4.4 Conclusion . . . 80

5 IMS-based distributed multimedia conferencing services for Next Generation Mobile Networks 83
5.1 Introduction . . . 83
5.2 3GPP IMS-based conference architecture . . . 84
5.3 Proposed IMS-based distributed video conferencing service . . . 85
5.3.1 Inter-connectivity with LTE/WiMAX networks . . . 85
5.3.2 Design requirements . . . 86
5.3.3 Proposed protocol of the IMS-based LTE/WiMAX distributed conferencing service . . . 87
5.4 Prototype and performance evaluation . . . 93
5.4.1 Evaluation method . . . 93
5.4.2 Evaluation results . . . 95
5.5 Conclusion . . . 96

6 Conclusion 99

Bibliography 103

Publications
Publication as first author:

- 1. Tien Anh Le, Hang Nguyen, "Application-aware cost function and its performance evaluation over scalable video conferencing services on heterogeneous networks", in proceedings of IEEE Wireless Communications and Networking Conference: Mobile and Wireless Networks 2012 (IEEE WCNC 2012 Track 3 Mobile Wireless), Publication Year: April 2012, Page(s): 1 - 6, .
- 2. Tien Anh Le, Hang Nguyen, Noel Crespi, "IMS-based distributed multimedia conferencing service for LTE", in proceedings of IEEE Wireless Communications and Networking Conference: In 2012 IEEE Wireless Communications and Networking Conference: Services, Applications, and Business (IEEE WCNC 2012 Track 4 SAB), Publication Year: April 2012, Page(s): 1 - 6, .
- 3. Tien Anh Le, Hang Nguyen, "Performance Evaluation of Application-Aware Cost Function for Scalable Video Multicast Streaming Services on Overlay Networks", Accepted in PIMRC 2012-Track 4:Services, Applications, and Business.
- 4. Tien Anh Le, Hang Nguyen, "Perception-based Application Layer Multicast Algorithm for scalable video conferencing", in proceedings of IEEE GLOBECOM 2011 - Communication Software, Services, and Multimedia Applications Symposium (GC'11 - CSWS), Publication Year: December 2011, Page(s): 1 - 6.
- 5. Tien Anh Le, Hang Nguyen, Hongguang Zhang, "Multi-variable cost function for Application Layer Multicast routing", in proceedings of 2010 IEEE Global Telecommunications Conference (GLOBECOM 2010), Publication Year: 2010, Page(s): 1 - 5.
- 6. Tien Anh Le, Hang Nguyen, Hongguang Zhang, "EvalSVC - An evaluation platform for scalable video coding transmission", in proceedings of 2010 IEEE 14th International Symposium on Consumer Electronics (ISCE), Publication Year: 2010, Page(s): 1 - 6.

- 7. Tien Anh Le, Hang Nguyen, Hongguang Zhang, "Scalable Video Transmission on Overlay Networks", in proceedings of 2010 Second International Conferences on Advances in Multimedia (MMEDIA), Publication Year: 2010, Page(s): 180 - 184.
- 8. Tien Anh Le, Hang Nguyen, "Centralized and distributed architectures of scalable video conferencing services", in proceedings of 2010 Second International Conference on Ubiquitous and Future Networks (ICUFN), Publication Year: 2010, Page(s): 394 - 399.
- 9. Tien Anh Le, Hang Nguyen, "Toward building an efficient Application Layer Multicast tree", in proceedings of IEEE-RIVF 2010 International Conference on Computing and Telecommunication Technologies (RIVF), Publication Year: 2010, Page(s): 1 - 6.

Publications as co-author:

- 1. Hongguang Zhang, Genet, M.G., Zeghlache, D, Tien Anh Le, Hang Nguyen, Crespi, N., "Collaborative home media community with semantic support", in proceedings of 2010 IEEE International Conference on Multimedia and Expo (ICME). Publication Year: 2010, Page(s): 1387 - 1392.
- 2. Hongguang Zhang, Hang Nguyen Hang, Noel Crespi, Shanmugalingam Sivasothy, Tien Anh Le, Djamal Zeghlache, Hui Wang, "A Metadata-Based Approach for Multimedia Service Mashup in IMS", in proceedings of 2010 Eighth Annual on Communication Networks and Services Research Conference (CNSR). Publication Year: 2010, Page(s): 356 - 360.
- 3. Mengke Hu, Hongguang Zhang, Tien Anh Le, Hang Nguyen, "Performance evaluation of video streaming over mobile WiMAX networks", in proceedings of 2010 IEEE GLOBECOM Workshops (GC Wkshps), Publication Year: 2010, Page(s): 898 - 902.
- 4. Quoc Tuan Tran, Tien Anh Le, Hang Nguyen, "WiMAX-based Overlay Conferencing Service", in proceedings of 15th International Conference on Intelligence in Next Generation Networks (ICIN) - From Bits to Data, From Pipes to Clouds (ICIN 2011), Publication Year: 2010, Page(s): 1 - 6.

Chapter 1

Introduction

Video is foreseen to be the next popular digital communication after voice communications [1]. Video conferencing service, the most complex type of video communications has been applied for a long time in business and everyday life. A multimedia conferencing service comprises of real-time, many-to-many communication sessions in which participants multicast their video, audio, exchange files, photos, and share their "white-board". A video conferencing session usually requires [2]:

- A control plane including session set up, modify, tear down,
- A data plane including transportation of media data, control and management of media mixers and media connections, and
- Video/Audio codec and security technologies.

To solve the centralized network architecture and heterogeneous network conditions problems, we will mainly concentrate on the scalable video codec and the data plane for media distribution. The control plane, audio codec, and security are out of this research's scope. However, for consistency, they will be explained when necessary.
Among the many existing problems of building a data plane for media distribution of scalable video contents, scalability is the main problem that conventional video conferencing services are facing and it is also the main problem that we are trying to tackle in this thesis. It can be the **terminal scalability** problem of encoding the video source once and consuming it at different types of terminals with a variety of screen sizes and computational capacity. The problem can simply be a cost function on which the **scalable overlay network** can be constructed to effectively multicast the scalable video contents. It can also be the **architecture scalability** problem that is preventing the conventional multimedia conferencing services from providing an effective service for a large number of participants. Last but not least, it can be the **scalable connectivity** problem on how the multimedia conference service can support terminals from a heterogeneous network with both fixed and wireless infrastructures. In this thesis, the above problems will be considered from different aspects. Within the available time frame, we have managed to solve the most part of the scalability problem of the conventional multimedia conferencing service that we have realized.
Regarding the **terminal scalability**, a multi-layer scalable video coding is suitable. H.264/MPEG-4 Advance Video Coding (AVC) [3] is a very famous video compression

standard. Its compression ratio has enabled many video communication services (such as video conferencing, video surveillance or video-phony...). However, a fatal limitation of this standard is that it is not scalable enough for many services. Once a source video stream has been encoded with AVC, that encoded bit-stream will remain the same throughout the communication process. Encoding parameters of the encoded bit-stream (such as bit-rate, frame-rate, screen size, SNR...) will be determined at the beginning of the communication session by senders and receivers (mostly by receivers). However, those senders and receivers may have different screen sizes, different computational capabilities, network conditions (such as bandwidth, delay, jitter...) which might be changed during the communication session. In those cases, in order for the AVC encoded bit-stream to be consumed adaptively at each and every receiver, there must exist middle-boxes in the communication network to convert the incoming AVC encoded bit-streams into various output bit-streams which are suitable for each receiver. This causes a huge delay in the communication session and single points of failure in the communication network. Otherwise the bit-stream will be stuck at bottle-necks and the entire video communication session will be broken. All of these problems make AVC not scalable enough for many video communication services.

People are now working and entertaining in a "3-screen" world. These screens are different in their computational capacities, screen resolutions, and communication bandwidths. A much better solution than AVC is to use Scalable Video Coding (SVC). SVC has been standardized as an extension of the AVC standard since 2007 [4]. The main idea of this extension is to apply multi-layer coding into the AVC codec. This is not a totally new idea since people had attempted to implement this idea from prior international video coding standards such as H.262/MPEG.2 Video, H.263 and MPEG.4 Visual [4]. However, the most challenging problem is that, the scalability used to come with a huge increase in the computational complexity. SVC has succeeded in providing scalability at an affordable computational cost. SVC encodes an input video stream into a multi-layer output bit-stream comprising of a base layer and several enhancement layers. Within those layers, the base-layer is encoded with a basic quality to guarantee that it can be consumed by the weakest receiver of the entire communication group. This base-layer is usually protected while being transmitted over the network by QoS assured transmission methods or Forward Error Correction (FEC) algorithms. For the purpose of backward compatibility, the base-layer must be recognized by all conventional H.264 decoders. Enhancement layers, when received at the receivers together with the base-layer, can enhance the overall-quality of the bit-stream. Especially, when all enhancement layers are received in-order at the receiver together with the base layer, the bit-stream will achieve its original encoded quality. However, when real conditions (such as bandwidths, delays, or displaying screen sizes) do not allow, upper layers can be discarded along the transmission link or at any middle box (relaying entities) for the bit-stream to be fit-in with those conditions without corrupting the video communication session.

Video services using SVC have been launched since the standardization of the SVC codec. SVC codec is dedicated to improve the transmission performance and not the encoding

performance of the video contents. However, few works have been conducted on the end-to-end transmission performance evaluation of SVC contents. In order to evaluate the end-to-end transmission performance of SVC video contents, designers and researchers are really in-need of a video transmission evaluation tool which is specially tailored for the evaluation of SVC encoded video transmissions over a real or simulated network. So far, the research community depends on Evalvid[5] for measuring the evaluation of AVC content transmission. Evalvid can only support up to the H.264 video codec. It cannot take SVC video as a possible input to the evaluation process. Moreover, Evalvid is limited in its interface to only real and Network Simulator (NS-2) based network environment. Evalvid's outputs, limited to only two metrics: Peak Signal to Noise Ratio (PSNR) and Mean Opinion Score (MOS), do not reflect well the performance of the video transmission. In our new EvalSVC platform, we manage to overcome all of these limitations by supporting the evaluation of SVC transmission. In addition to the conventional metrics of PSNR and MOS, we also use Structural Similarity Index (SSIM). SSIM is a method for measuring the similarity between two images. It was designed to improve the traditional methods like Peak Signal-to-Noise Ratio (PSNR) and Mean Squared Error (MSE), which have proved to be inconsistent with human eye perception. In our new EvalSVC platform, more interfaces with the bottle neck and overlay networks are developed. Since bottlenecks can be found everywhere on the network and it is actually a serious problem for the multimedia services, the interface to the bottle-neck network is really necessary for SVC transmission evaluation. The interface to the overlay network is required because overlay network is the future transportation architecture of the multimedia services (such as peer-to-peer communication and cloud-based multimedia services).

EvalSVC is an evaluation platform for SVC content transmission. Designed with different interfaces to both real and simulated networks, EvalSVC can be used to evaluate the transmission performance of SVC contents over different network environments. It is a need for the industrial and research communities because it can help the researchers and SVC-based service developers to evaluate the performance of their services and fine-tune them from different quality metrics. This EvalSVC tool is of great interest for the industrial and research communities because it is built and distributed as an open-source platform (http://code.google.com/p/evalsvc/) so that the researchers and developers can freely investigate, use, develop, and enrich the platform based on their requirements and needs. To reach the purpose of evaluating the SVC contents transmission, EvalSVC has the functionalities of a real SVC video streaming server. It can be fully integrated with different network environments using both real video dumping files and popular network simulation platforms such as the Network Simulator (NS-2). We even have interfaces to Oversim, an overlay simulation platform and a hybrid transmission environment between simulated and real networks to support richer evaluation scenarios [6, 7]. The input of the EvalSVC platform can be all different types of SVC-encoded contents (temporal, spatial, SNR scalability) as well as AVC encoded video contents. After a transmission over the network defined by the network environment, the output results can be the

objective QoS-related parameters of the under-layer networks (such as loss-rate, delays, jitter...) and three different subjective and objective metrics. The subjective metric is the Mean Opinion Score (MOS). The objective metrics are Peak Signal to Noise Ratio (PSNR) and the Structural Similarity Index (SSIM). Users can judge the performance of the video content delivery based on a single metric, any combination of them or all of the metrics. An evaluation session using EvalSVC starts with the raw video taken from a file or real-time captured by a camera. This raw video will then be encoded by the SVC encoder to form a SVC bit-stream. The SVC encoded bit-stream is packetized into RTP packets accompanied by a hint track. EvalSVC will stream the hinted video contents from the Sender node to the real/simulated network. A video trace file, a sender and a receiver dumping files will be generated. Using information from all of these files, and the original bit-stream, the SVC Re-builder at the receiver's side will reconstruct the received bit-stream and feed it to the SVC Evaluator for generating the video transmission results. The reconstructed SVC video can also be delivered to the SVC decoder to get the output video play-out at the receiver side.

In comparison with the conventional evaluation platform (such as EvalVid), the main advantage of the newly proposed EvalSVC is its capability to support SVC-encoded contents. Moreover, EvalSVC has enriched interfaces to the overlay simulation platform (built by Oversim). In this paper, we introduce also the EvalSVC's interfaces to real networks such as real Internet or real wireless networks (GPRS, UMTS, WiMAX, LTE, WiFi...), to a hybrid network environment, and to the Overlay simulation platform as the samples of possible interfaces to the platform. In [6], these interfaces to real networks and hybrid networks have been applied for evaluating SVC video transmission over a real mobile WiMAX network and a hybrid network composed of a simulated Internet topology and a real mobile WiMAX network. The real interfaces help EvalSVC to obtain real measurements of the video transmission session on real networks. The Oversim interface helps EvalSVC to simulate an overlay network environment so that distributed SVC-based video content delivery can be evaluated with our EvalSVC. More interfaces can be added to this open source platform upon needs and requirements of the research and industrial communities.

Since our EvalSVC platform is open to the public [8, 9], many people have used our EvalSVC platform for their research and development works. We have been contacted by many EvalSVC's users from both the academy and the industry. For example, we have received interesting questions and requests for support from industrial laboratories and companies such as NEC Labs. America-USA, CTI-Greece, and from universities such as UC Davis-University of California-USA, Kumamoto university-Japan, Lisbon University Institute-Portugal, ITS Surabaya and Universitas Nusa Cendana-Indonesia. We have received quite a high citation by quality external research works such as [10, 11, 12, 13, 14, 15, 16] as well as our college's research works such as [17, 18, 19, 20]. Regretting of not having enough resources to fully support the public's concerns about the EvalSVC platform, the first contribution of this thesis is to give a full and detailed descriptions of this tool-set so that people can handle EvalSVC, integrate it into their

research, enrich it upon their needs and requirements to make it a real community's platform for researchers and developers of SVC-based multimedia services.
The second contribution of this thesis is to conduct the performance evaluation of SVC contents end-to-end transmission over different network environments. SVC is dedicated to improve the performance of video transmission. Thus, the purpose of the second part of this thesis is to verify on realistic network conditions such as bottleneck conditions and overlay networks whether SVC end-to-end transmission really archives its objectives of improving video transmission. The conventional evaluation of SVC and AVC encoding schemes were only to compare AVC's and SVC's source encoding performance. Choosing a different approach from the conventional research works on the evaluation of SVC encoding performance, we manage to provide an evaluation platform for SVC transmission with the availability of EvalSVC. We can now evaluate the entire end-to-end transmission of SVC contents on different types of network conditions. It means that, even though in theory, we can partly understand that SVC should be better than AVC with its scalability feature at the encoding step, but it is only through the performance evaluation results and metrics provided by the EvalSVC platform that we can confirm if the SVC end-to-end transmission is really better than AVC end-to-end transmission in different network conditions.
EvalSVC is applied to evaluate the video services in different scenarios. In the first scenario, we want to compare the end-to-end transmission performance of AVC and different types of SVC scalability contents in a bottle-neck network environment. This evaluation is new because it is a transmission evaluation of different types of SVC scalability and AVC in a bottle neck environment and not their encoding evaluation. A bottle neck can be defined as the difference between the core network's bandwidth (usually with broadband links) and the available bandwidth of the users' terminals (usually limited or narrow bandwidth link). Sometimes, due to the limited computational capacity or a small buffer memory at the terminal, the multimedia applications running on that terminal have to suffer an even smaller bandwidth than the maximum available bandwidth of the users' terminals. This is a very common and serious problem for multimedia services because of the asymmetric bandwidths between the core and the access networks. It is getting worse because users are using different types of terminals (computers, tablets, smart phones) and these terminals increase the asymmetric bandwidths between the core network and the access terminals/networks. A big multimedia flux coming from a high bandwidth link can not go through narrow bandwidth link. Thus, congestion can happen and the encoded video contents may be blocked on the bottlenecks (the conjunction between the high-bandwidth link and the narrow bandwidth link) before they can arrive to the receivers, creating delay, jitter, packet loss... We select this bottle-neck condition because it is a very common problem of today's network and multimedia services. By evaluating the performance of AVC and different types of SVC scalability content delivery over a bottle neck network, we want to find out if AVC actually has a big problem with congestions in the network and whether SVC can be a solution to this problem. If SVC transmission is better, the purpose is to find which type of SVC scalability would

be the best. We choose to evaluate the multimedia services under a bottle neck effect because it is very common in real networks. To evaluate the scalable capability of the SVC contents in a bottle neck network, we use EvalSVC to evaluate the SVC transmission performance against the very popular conventional AVC transmission.
In the second scenario, we evaluate the transmission performance of AVC and different types of SVC scalability contents (temporal, spatial, SNR scalability) in an overlay network environment. This evaluation is original because it is the transmission evaluation of AVC vs. different types of SVC scalability, and not their encoding evaluation. Overlay network is a computer network which is built on the top of another network. For example, distributed systems such as cloud computing, peer-to-peer networks are overlay networks because their nodes run on top of the Internet. Since this type of network is the future transportation architecture of the multimedia services, the purpose for us to select it as a network setting for evaluating the performance of different types of scalability of SVC contents is to find out whether SVC transmission is really better than AVC transmission over the overlay network. And if SVC transmission is better, which type of SVC scalability would be the best suitable for the SVC-based multimedia content delivery via the overlay networks, the future of distributed multimedia delivery.
For the cost function on which the **scalable overlay network** can be built, we propose an application-aware multi-variable cost function. Multicast is the method of delivering data over a group of pre-registered destinations. Conceptually, multicast has a better performance than both unicast and broadcast when serving only a certain group of users. The Internet was originally built for unicast or one-to-one applications. Nowadays, it has to serve a large number of multimedia services such as multimedia conference or multi-player games. These types of multicast services put a big load on the unicast infrastructure of the Internet. Therefore, there is a demand for the design and deployment of multicast algorithms over the Internet. Multicast can be approached from either network layer or application layer. Regarding multicast in network layer, IP-Multicast[21] is the first attempt to solve this problem. It is so far the most efficient multicast mechanism for delivering data over each link of the network only once. However, many deploying problems are still preventing IP-Multicast from being supported worldwide[22]. Attempts have been made to overcome these problems. Explicit Multi-Unicast (XCAST)[23] is an alternate multicast strategy to IP multicast that provides reception addresses of all destinations within each packet. As such, since the IP packet size is limited in general, XCAST cannot be used for multicast groups of large number of destinations. An alternative solution for Multicast is to build Application Level Multicast(ALM) over the underlay network. The key concept of ALM is the implementation of multi-casting functionality as an application service instead of a network service. It has excellent advantages over IP-Multicast: easier and possibly immediate deployment over the Internet without any modification of the current infrastructure and adaptable to a specific application. Tree-push is a common approach for data delivery in Application Layer Multicast algorithms, especially when the multimedia quality is concerned[24]. In this approach, before the data distribution can take place, a media distribution tree

must be built from all participating peers, then the data is actively distributed from the source node to intermediate peers until reaching all peers in the multicast tree[25]. In order to build that ALM distribution tree, we must have costs of all available end-to-end links. Those costs can only be calculated by using a cost function. The construction algorithm of the media distribution tree is based on the costs among participating peers. Eventually, the efficiency of the media distribution algorithm will mainly depend on the cost function being used. The proposed cost function should not be confused with conventional network layer-based cost functions since they are not working on the same layer. In fact, application layer cost function relies on network layer cost function for underlay's routing and only be responsible for routing on the overlay level of Application Layer Multicast.

The conventional cost functions are mainly based on classical network layer cost functions. Moreover they are either single variable or heuristic. In this thesis, we propose a cross-layer multi-variable cost function to reflect the fact that, the applications and the networks have many different QoS parameters (bandwidth, delay, Packet Error Rate, Bit Error Rate, jitter...) and a cross-layer multi-variable cost function can better model the real network's available resources than a single variable cost function. Our cross-layer multi-variable cost function makes an optimization simultaneously over several QoS parameters, both from the application and the network layers; whereas conventional single variable cost functions are all optimizations for only one single QoS parameter. If a single-variable cost function can only obtain a local optimal value of the cost, our cross-layer multi-variable cost function can generate a bigger set of routes based on their multi-dimensional cost values. Therefore, the ALM algorithm can search and find a better optimum route from the bigger route set. Nowadays, the multimedia application has dynamic resource requirements. In order to support that, the newly proposed cost function considers both the requirements from the application layer and the available resources from the network layer to make a cross-layer optimization. This multi-variable characteristic and this cross-layer optimization make our proposed cost function more suitable for today's enriched and complexed multimedia services. A theoretical model has been built in order to obtain the exact mathematical expression of the cross-layer multi-variable cost function. The mathematical derivation of our cost function makes it extensible and applicable in more general cases of different contexts. All the steps of this mathematical derivation are also described in details to be further applied to find other forms of cost functions with other QoS parameters. For example, our proposed mathematical derivation and theoretical model can be applied to get the cost functions for different types of services with any arbitrary number of QoS requirements (even for $N \geq 3$).

Because of its multi-layer characteristics, Scalable Video Coding (SVC) is a very suitable video codec for multimedia applications designed to work on a convergent network. Besides, SVC contents are very suitable for multimedia applications such as multimedia conference or multi-player games because of their advantages over conventional video coding methods[7]. Scalable Video Coding is the multi-layer extension of Advanced

Video Coding with the advantage of providing visual services for customers with convergent network conditions and terminals' capabilities. SVC-based multimedia applications dynamically change their QoS requirements during the multicast session and this can be supported very well with our newly proposed Application-Network cross layer multivariable cost function. More specifically, video conferencing and video multicast services are the typical multimedia applications whose resource requirements are different. For example, a video conferencing service requires a small delay and a quite high bandwidth while video multicast may have a lower strain on the delay but the required bandwidth may be higher. In this research, we evaluate interesting and useful multimedia services built on scalable video coding and our newly proposed cost function. These services are the models for other one-to-many and many-to-many multimedia services. They can support convergent context of terminals with different available bandwidth, different screen-sizes and a variety of computational capacities. Our contributions are also to evaluate popular SVC-based services built on our newly proposed cross-layer multivariable cost function. They are a scalable video coding-based multicast service built on an overlay network as a model for many video streaming or one-to-many services, and a scalable video coding-based conferencing service built on a convergent underlay network as a model for many many-to-many multimedia services (such as video conference or multi-party multimedia games).
Wireless network is a very important part of telecommunications. With the coming of 4G wireless networks such as LTE and WiMAX, broadband wireless communication becomes a reality. WiMAX is an IEEE (Institute of Electrical and Electronics Engineers) specification also known as IEEE 802.16. Its most famous releases are IEEE 802.16d (2004), IEEE 802.16e (2005) and IEEE 802.16m (2009). LTE is a 3GPP (Third-Generation Partnership Project) standard. It has two main release groups: LTE and LTE-Advanced. Multimedia services are the killer applications of 4G wireless networks. Few or almost no research work has been found on the evaluation of cost functions and ALM algorithms on a wireless network in general and a real wireless network in particular. Convergent network comprised of Internet and wireless network is the current trend of telecommunications. Users can use different types of terminals and network infrastructures to connect to the multimedia services such as video conferencing, multi-party games, and video streaming. For example, in a same video conference, some users can participate to the conference from their smart phones using a mobile WiMAX wireless network whereas other users can participate from their fixed terminals through the Internet. WiMAX (Worldwide Interoperability for Microwave Access) IEEE 802.16e 2005 release is one of the 4G wireless communication technologies for delivering high-speed Internet service to large geographical areas. Here we use this IEEE 802.16e release which is also called "Mobile WiMax" release. We have conducted evaluations of multimedia services on a real WiMAX mobile network provided by POSEIDON, a French National Pole de Competitivite System@tic's project. We have evaluated the SVC-based multimedia services on an overlay network constructed from our newly proposed cost function. The overlay network is built over a convergent network of the simulated Internet topology

and the real mobile WiMAX network. The available mobile WiMAX network comprises of a real access network and a real core network. The mobile WiMAX access network comprises of an Acatel-Lucent extended Base Station. The core network comprises of the Operations and Maintenance Centre, the Wireless Area Controller, the Authentication, Authorization, and Accounting, the Home Agent, the Dynamic Host Configuration Protocol, FTP and video servers. We also have two types of real WiMAX mobile terminals. The first type of WiMAX mobile terminal is an Alcatel-Lucent 9799 PCMCIA card. The second type of WiMAX mobile terminal is a Sequans USB card. Together, the dynamic multimedia applications and the convergent network comprised of a simulated Internet topology and the real mobile WiMAX network make an ideal platform for us to evaluate the performance of our cross-layer multi-variable cost function.
We are now able to evaluate performance of our newly proposed cost function for the SVC-based services (scalable video multicast and scalable video conferencing) on a convergent network of the simulated Internet topology and the real 4G mobile WiMAX network.
Our proposal is original because of the cross-layer multi-variable characteristics of the cost function. But it's also the scalable video coding-based multicast and video conferencing services and their performance evaluation results on the overlay network built over a convergent underlay network comprised of a simulated Internet topology and a real mobile WiMAX network that make our proposal unique. In this research, all of these requirements are for the very first time integrated and evaluated on a common platform to show the performance of the newly proposed cross-layer multi-variable cost function on the popular SVC-based services built over an overlay network whose underlay network is the convergence of a real mobile WiMAX network and a simulated Internet topology.
In brief, we have the following contributions:

- Propose a new cross-layer multi-variable cost function taking into account both the requirements from applications and the available resources from the networks,
- Propose in details the mathematical derivation process of the new cross-layer multi-variable cost function and analyze its performance. This analysis and derivation process can be further applied to find other forms of cost functions with other QoS parameters,
- Perform theoretical analysis to compare the newly found cost function with conventional ones,
- A scalable video coding-based multicast service built on an overlay network. This can be a model for many video streaming or one-to-many services,
- Evaluation of the new cost function in comparison with the conventional cost function over an overlay network in a scalable video coding-based multicast service setting,

- A scalable video coding-based conferencing service built on a convergent underlay network. This can be a model for many many-to-many multimedia services (such as video conference or multi-party multimedia games),

- Evaluation of the new cost function in comparison with the conventional cost function over an overlay network built on a convergent underlay network of a real mobile WiMAX network and a simulated Internet topology in a scalable video coding-based conferencing service setting,

Architecture scalability can be referred as the capability of scaling the conferencing architecture to support different numbers of participants without modifying too much and too frequently the supporting structure. A very common conventional multimedia conferencing architecture is to use a Multipoint Control Unit (MCU) as the central point for receiving, trans-coding, trans-rating, mixing, and then sending back to participants (centralized service architecture). However, the cost of conventional conferencing method increases quadratically with the increasing number of participants. It cannot support well users with different types of terminals due to scalability problems. Moreover, the centralized architecture has many disadvantages such as cost (incapable of decreasing neither Capital Expenditures-CAPEX nor Operational Expenditure-OPEX) or a high delay especially when the number of participants increases. Although the signaling plan with P2PSIP has been proposed and developed in the research community[26], the overall distributed architecture multi-party communication is however not yet mature. The advantage is that it aims at distributing the computational loads of the conference session to all participants based on their capabilities. Centralized and distributed architectures have their own advantages and questions are usually asked about which architecture is preferable in certain network and usage conditions. More particularly, is there any distributed architecture that can outperform the centralized architecture in terms of system's delay and total cost? What is the optimal method to group the peers in a distributed architecture? How to solve the problem of the excessive signaling and data forwarding messages in the distributed architecture to save mobile terminals' battery and computational capacity? Conventional methods will be analyzed based on these questions and concerns. Conventional multi-party video conferencing service begins with centralized solutions. Main players in this centralized architectures are Cisco (with Tele-Presence)[27], Polycom[28]. These players concentrate on building High Definition video conferencing services based on MCU. Vidyo is the latest player in the video conferencing market with the advantage of using Scalable Video Coding in the service[29]. Though, the centralized architecture with MCU is still applied. Microsoft Office Live Meeting (Professional User License) uses a Server/Client architecture and has the ability to schedule and manage meetings with up to 1,250 participants. However, only few participants can be presenters who can upload their videos and the others are non-active attendees[30]. In[31], the centralized service architecture of conventional centralized conferencing services was combined with Scalable Video Coding (SVC) in order to provide it with more scalability. A new MCU architecture (Scalable Video

Conferencing Server-SVCS) was proposed. Similar to the conventional centralized architecture, SVCS received SVC bit-streams from all participants. Nevertheless, it directly routes bit-streams to its desired destinations without any mixing, trans-coding/rating. All the workload will be processed at participants' terminals. This is so far one of the most advanced methods of centralized scalable video conferencing service which is widely deployed because of its scalability and advanced performance when compared to the conventional centralized methods. When more participants want to join the conference (e.g. at big events), the cost of the centralized architecture increases sharply. Therefore, distributed architectures have been proposed. Skype, a peer-to-peer VoIP client developed by KaZaa in 2003, has so far the most popular Internet-based video conferencing service. This application is able to throttle its sending rate to match the unpredictable Internet bandwidth while preserving resource[32]. Like its file sharing predecessor KaZaa, Skype is an overlay peer-to-peer network. There are two types of nodes in this overlay network, ordinary hosts and super nodes (SN). An ordinary host is a Skype application that can be used to place voice calls and send text messages. A super node is an ordinary host on the Skype network[33]. Its main limitation is because the "super node" architecture requires an infrastructure, similar to CDN (Content Delivery Network), to be built and maintained in order to provide the video conferencing service. Moreover, as a commercial production, all its architectures and protocols are closed-source. This brings many difficulties for the research community in their attempts to approach Skype system's performance. Spiers et. al.[34] implemented IP multimedia subsystem (IMS)-based video conferencing systems with two different architectures, Server/Client and P2P, and measured their signaling and data traffic overhead. The results show that Server/Client offers better network control together with a reduction in signaling and media overhead, whereas P2P allows flexibility, but at the expense of higher overhead. Web-based video conferencing services (such as Dimdim) are dominating the Internet, however, they are really limited in delays, discontinuities, errors and limited functionality[35]. Many other popular online chatting applications (like MSN, Yahoo messenger, Google talk, etc.) only support multi-party audio conference and 2-party video conference, and therefore are not considered. Naturally, when a multi-point communication service is to be provided on a distributed scheme, multicast should be involved as an essential approach to reduce bandwidth demands of multi party video conferencing. IP-multicast[36] is so far the most efficient type of multicast. However, deployment issues are preventing it from being widely applied[22]. An alternative approach to multi-party video conferencing is presented in[37]. The system is based on the use of a distributed peer-to-peer (P2P) architecture which does not need any special hardware or network infrastructure support. There is no additional networking and computing resources needed at the end points more than that of a point-to-point video conference. In this approach, each participant could see one other selected participant under most practical cases. There are cases where some participants cannot be seen by some others and the number of such cases increases with the number of participants. In[38], a distributed architecture has been proposed for video conferencing service based on an Application Layer Multicast

(ALM) algorithm. However, the proposal has not considered the scalability problem of terminals. Moreover, the cost function used to construct the media distribution tree has not been investigated. In[39],[40], scalability has been considered and SVC has been applied to support varied types of participants' terminals. Nevertheless, the proposed algorithms can only support a limited number of participants (e.g.10-15) since they do not have a mechanism to limit the unnecessary traffic on the multicast tree. In[41], Chu et al. proposed an End System Multicast architecture to support video conferencing applications, where multicast functionality is pushed to the edge. This mesh-based architecture requires a lot of signaling in order to maintain the full mesh among all peers. Lennox and Schulzrinne[42] proposed a full mesh conferencing protocol without a central point of control. Since each peer has its own capacity, if everyone has to contribute the same computation and resource into the mesh, the weaker peers may not be able to maintain their connection. This breaks the stability of the architecture. Luo et al.[38] proposed to integrate application layer multicast with native IP multicast in P2P conferencing systems. Although it is mainly based on P2P architecture, it fails to consider the limited capacity of the mobile terminals. Moreover, it does not consider the relation between the video codec and the video conference service architecture. Through[43], we know that it is possible to compress two statistically dependent signals in a distributed way (separate encoding, joint decoding) approaching the coding efficiency of conventional predictive coding schemes(joint encoding and decoding). When applied to video coding, Distributed Video Coding (DVC) is very promising. Video conferencing with mobile devices is a relevant application for DVC. Beside many advantages, DVC still has fatal disadvantages. The first and main one is that it has a high decoding complexity as one of the main DVC characteristics is the potential to shift the complexity from the encoder to the decoder. In current DVC approaches, the required decoding complexity seems to be rather high; in applications requiring real-time decoding, this is a significant drawback (that should become less relevant with time). The second drawback is its lower compression efficiency (until now, DVC did not reach the same level of compression efficiency as state-of-the-art predictive coding, notably the H.264/AVC standard). Therefore, despite of its interesting concept regarding the distributed video conference service, DVC is still not a direct competitor with SVC. In[44], the author proposes a distributed video conferencing service on heterogeneous networks with the support from just a normal PC standing outside of the overlay network and act as the MCU. While this solution is interesting, the normal PC has to carry on heavy computational load as if it is a MCU. This single point of failure can break the whole distributed service when the traffic gets high. From the best of our knowledge, none of the proposed distributed architectures has compared its performance with the conventional centralized architecture. Thus, no conclusion can be made on whether a distributed architecture is really better than the centralized architecture and whether service users should bother changing their current centralized conferencing system with a new distributed system. So far,[45] is the early attempt to compare the two scalable conferencing architectures using its proposed simulation platform[46]. The simulation results show that, when using

a much higher computational capacity, the MCU of the centralized architecture can only provide a similar bit-rate with the distributed architecture at the trade-off of a much higher delay. The results were interesting but they need to be validated by theoretical analysis in order to be applied in more general conditions. To solve the problem of the architecture scalability of the multimedia conferencing services, we have the following contributions:

- **An enriched human perception-based distributed video conferencing architecture is proposed** in which limitations of human perception when participating into the conference will be considered in order to reduce the unnecessary traffic on the overlay network,
- **Theoretical queuing models** are constructed and analysed for the four main performance criteria: total waiting time, end-to-end delay, computational requirements and total throughput of the three different architectures: centralized perception-based centralized and the proposed perception-based distributed architectures. The performance of the centralized, perception-based centralized and perception-based distributed architectures have been considered and compared in terms of waiting time, end-to-end delay, required service rate, and total throughput. Mathematical analysis and expression for these parameters have also been constructed. The waiting time performance compares the processing and queuing mechanisms in the three architectures. The end-to-end delay comparison considers the delay of a packet including the processing time in the queues and also the transmission time in the underlay network. The required service rates explain how much computation capacity the important nodes have to equip in order to support steady states of the queues formed and processed by each architecture. Finally, the total throughput performance answers very common concerns of whether the distributed architecture usually has much more traffic than the centralized architecture or not. By approaching the performance from different aspects, a completed view of the proposal is archived.
- **Numerical simulations of the analysing model** Computation of the analysing models for main performance criteria such as total waiting time, end-to-end delay, computational requirements and total throughput for centralized, perception-based centralized and perception-based distributed. Regarding all the above parameters of the three architectures, the perception-based distributed architecture performs better than the other two in all four main aspects of performance.

The performance of the centralized, perception-based centralized and perception-based distributed architectures have been considered and compared in terms of waiting time, end-to-end delay, required service rate, and total throughput. Mathematical analysis and expression for these parameters have also been constructed. The waiting time performance compares the processing and queuing mechanisms in the three architectures. The end-to-end delay comparison considers the delay of a packet including the processing time in the queues and also the transmission time in the underlay network. The

required service rates explains how much computation capacity the important nodes have to equip in order to support steady states of the queues formed and processed by each architecture. Finally, the total throughput performance answers very common concerns that whether the distributed architecture usually has much more traffic than the centralized architecture or not. By approaching the performance from different aspects, a completed view of the proposal is archived. Regarding all the above parameters of the three architecture, the perception-based distributed architecture performs better than the other two in all four main aspects of performance.

Another bottle-neck of a scalable conferencing service is its **scalable connectivity** or the possibility of providing the scalable conferencing services for users participating into the conferencing session from different types of infrastructures (such as wireless or wireline). As explained previously, the distributed architecture is the future of multi-party, multimedia conferencing services. In terms of delay performance and required computational capacity, distributed architecture out-performs its centralized counterpart as shown in [47]. The quality of video content transmitted over the specifically designed overlay network has also been proved to be better than that of centralized architecture, using objective quality evaluation methods[45]. One of the main reasons why distributed architecture has not been widely applied in the wireless networks is that it increases the required bit rates at mobile terminals and within the network. Indeed, in mobile wireless networks, bandwidth is a very costly and limited resource. Therefore, it is almost impossible (for users) to apply directly the basic distributed architecture in mobile networks (due to the extra requirements in bit-rates, mobile terminals' battery life, computation and the wireless resources for maintaining the overlay structure). To date, the mobile network and services have only been based on a centralized architecture. Almost no work has been found for distributed video conferencing service on mobile networks. Recently, mobile participants have been equipped with high computational devices using radio access networks (e.g. 4G: Long Term Evolution, WiMAX), bigger screen sizes and better computational capacity. Real time multimedia services (such as video conference, video streaming) are foreseen to be "killer"-applications on 4G. It is expensive and difficult to provide such services based on a centralized architecture. However, the distributed video conferencing service architecture and the 4G network architecture have been designed separately without considering the other's requirements. Therefore, it is currently also difficult to provide real-time multimedia services based on a distributed architecture over 4G networks. 4G networks rely on the IP Multimedia Subsystem (IMS[48]) to provide multimedia services including video conferencing. In turn, the IMS-based video conferencing service is built and standardized for MCU-based or centralized service architecture. Therefore, it shares many similar problems with the centralized video conferencing service architecture[45], especially when the number of User Equipment (UE) units that participate in the conference via the 4G infrastructure increases. This thesis proposes a solution for a distributed architecture that allows the reduction of bit rates required by peers to save their valuable radio resources and make the distributed architecture possible for next generation mobile networks. In this thesis, the LTE/WiMAX network is

used as a demonstration of a 4G infrastructure. Due to the many advantages of the distributed architecture over the centralized architecture, the main purpose of our research is not to compare it with the centralized architecture. That is out of the scope of this research and we leave it to the many other ongoing research projects. Our purpose is to enable the distributed architecture video conferencing service for next generation mobile networks with slight modifications of the 4G infrastructure and of the IMS platform. To enable the distributed architecture to function over a mobile network, we have to solve the crucial problem of the high bit-rates required at the wireless terminals. In our solution, we will try to overcome this main disadvantage of the distributed architecture, which is why we will evaluate the performance of our proposal in terms of bit-rates for data and signaling plans. A new solution is proposed here, which makes it possible for the current IMS-based LTE/WiMAX infrastructure to seamlessly support distributed video conferencing services. The main contributions of the research are to:

- Propose an IMS-based architecture that supports LTE/WiMAX's UEs and WiMAX's SU to participate in distributed scalable video conferencing service without using a centralized MCU. It significantly reduces the bit-rates required at mobile terminals, thus conserving the wireless resources,
- Develop a proof-of-concept prototype to prove the feasibility and compatibility of the newly proposed solution and
- Evaluate the performance of the proposed system under audio and video conference working scenarios.

Our proposed architecture's requirements call for:

- New SIP messages and some standard ones with modifications for new functionalities, new purposes and in new contexts in their destinations and content,
- A minimum modification of the 4G's BS(s),
- New xAS features.

Throughout the chapters in this thesis, the problems will be further investigated and solutions will be provided based on both theoretical analysis, evaluation results, prototypes and real 4G wireless network infrastructures.

Chapter 2

End-to-End Transmission of Scalable Video Contents: Performance Evaluation over EvalSVC-A New Open Source Platform

Contents

2.1 Introduction 19
2.2 Proposed EvalSVC Platform Description 20
2.2.1 Architecture description 20
2.2.2 Functional blocks 21
2.2.3 Evaluation metrics 23
2.3 Proposed EvalSVC's interfaces 24
2.3.1 Proposed interface to real networks 24
2.3.2 Proposed interface to Overlay simulation platform 25
2.3.3 Hybrid network environment 25
2.4 Performance comparison of AVC vs. different types of SVC end-to-end transmission on EvalSVC in a bottleneck network environment 26
2.5 Performance comparison of AVC vs. different types of SVC end-to-end transmission on EvalSVC over an overlay network environment 29
2.6 Conclusion 30

2.1 Introduction

Scalable Video Coding (SVC) is the latest extension of the famous Advance Video Coding (AVC) standard. Scalability is important and useful because it is dedicated to the

transmission of video contents over heterogeneous network conditions and terminals' capabilities. Nevertheless, the multimedia service research community and industry have not been able to fully utilize the entire potential of this video coding standard extension because of the lack of an evaluation platform for evaluating the end-to-end transmission of SVC-contents. EvalSVC aims to foster SVC-based applications and research in multimedia services. It is capable of evaluating the end-to-end transmission of SVC bit-streams encoded with enhanced features (spatial, temporal, SNR, and combined scalability). The output results are both objective and subjective metrics of the video transmission. Interfaces with real networks and an overlay simulation platform are presented. Through these interfaces, the transmission performance of different types of SVC scalability and AVC bit-streams on a bottle-neck and an overlay network will be evaluated. This evaluation is new because it is conducted on the end-to-end transmission of SVC contents and not on the coding performance. The bottle-neck network environment is set up for evaluation because this realistic and very common network condition usually causes serious problems for video transmission and multimedia services. The overlay network is used for evaluation because it is a future transportation environment for multimedia content delivery. The results show that, both in bottle-neck network conditions and on the overlay network environment, SVC transmission outperforms AVC transmission and SNR scalability has the highest transmission performance. Through these end-to-end transmission performance evaluations over EvalSVC, we can conclude that, indeed SVC achieves its objectives of improving video transmission over realistic network conditions.

2.2 Proposed EvalSVC Platform Description

2.2.1 Architecture description

Our work manages to develop a video transmission evaluation framework supporting SVC's NALU extension types. The Network Abstraction Layer (NAL) is the interface between the SVC encoder and the real network protocol, which will be used to transmit the encoded bit-stream. The NAL encoder encapsulates the output slices from the VCL encoder into Network Abstraction Layer Units (NALU), which are suitable for transmission over packet networks or used in packet oriented multiplex environments [49]. In order to generate proper NAL units, we must pre-define the network protocol that we want to use to transmit the video bit-stream. H.264/AVC and SVC support encapsulating VCL slices into a number of network protocol (H.320, MPEG-2, and RTP) [50] in which RTP is mostly used because of its popularity. The most difficult problem is that those extending types haven't been fully defined and standardized by IETF. In order to evaluate the SVC transmissions, it is required that Hinter and SVC Rebuilder blocks in Fig. 2.1 must support SVC new NALs so that the sender and receiver can packetize and rebuild the SVC frames. Therefore, these blocks are essential for SVC evaluation. However, it should be noticed that, the basic NALU extension types (e.g., types 14, 15, 20) have been spared for SVC extensions from AVC NALU types. So we are

going to support only those NALU extensions in our EvalSVC framework since they have already reflected the main concepts of SVC. Other NALU types, such as Payload Content Scalability Information (PACSI), Empty NAL unit and the Non-Interleaved Multi-time Aggregation Packet (NI-MTAP), which are being drafted in [49], are out of our scope. A NAL unit comprises of a header and a payload. In AVC, the NALU's header is 1 byte length [51]. Meanwhile, a SVC's NAL header can be 1, 2, or 3 octet length [52]. The first octet of SVC's NAL header is identical with AVC (Fig. 2.2). It contains 3 fields of which 2 first fields (F, NRI) are spared for signaling wire-line/wireless gateway, and the importance of that NALU. The last field in the first octet of the SVC's NAL header is *NALU Type* specifying the NAL unit payload type. NAL unit type 14 is used for prefix NAL unit, NAL unit type 15 is used for subset sequence parameter set, and NAL unit type 20 is used for coded slice in scalable extension. NAL unit types 14 and 20 indicate the presence of three additional octets in the NAL unit header. NALU types 15 contents header information which is not necessary to be repeatedly transmitted for each sequence of of picture [53]. This sub-sequence parameter set can be transmitted on an "out-of-band" transmission for error resilience. We will need this information about the NALU types when we reconstruct the possibly corrupted SVC bit-stream at the receiver side. PRID (priority ID) specifies a priority identifier for the NALU. A lower PRID indicates a higher priority. DID (dependence ID) indicates the inter-layer coding level of a layer representation. QID (quality ID) indicates the quality level of an MGS layer representation. TID (temporal ID) indicates the temporal level of a layer representation.

2.2.2 Functional blocks

Figure 2.1 illustrates basic components of our EvalSVC platform. Some external tools are also integrated into EvalSVC to support the data-flow of the entire framework.

- Raw video in: This is the input video. Normally the YUV or CIF formats are used as they are acceptable by SVC encoders as well as common video capturing devices.
- SVC encoder/decoder: We use JSVM [54] as our main SVC codec.
- ***Hinter***: This component is derived from the mp4box tool of the GPAC library [55]. The main role of this component is to packetize SVC's NALU into RTP packets and add a hint track to the SVC bit-stream. We can consider the hint track as an in-band signaling for the SVC bit-stream. Another option is to distribute the hint track in the format of a SDP file via a separate channel as out-band signaling. The difficulty that the SVC Hinter must support new SVC's NALUs as described in Fig. 2.2. Because these new NALUs are being standardized at the time we were developing the EvalSVC platform, therefore, it is difficult for us to support the right NALUs while still keep the possible extensions for other NALUs which may be standardized later. Nevertheless, the rapid development of SVC-based services requires the instant support of the Hinter on new NALUs. We have developed the

Hinter so that it supports the basic NALUs which are required for SVC transmission and reserves some units for possible extensions.

- Mp4trace: This component acts as a video sender. Its main part is to send the hinted SVC bit-stream out to the network using the packetization information it has from the Hinter. It also logs the sequence numbers, types, and sizes of the video frames, and the number of UDP packets used to transmit each frame (since the frame size may exceed the UDP/RTP maximum payload sizes), and its sending time-scale. Mp4trace can work in streaming mode or camera mode.

- Networks: 2 kinds of networks can be used in EvalSVC, real and simulated networks. Real network's conditions can be obtained by using real IP connections over the Internet. Tcpdump can be used to trace the real network traffic at both ends and to form the sender's and receiver's dumping files. We can also use NS-2 simulated network to form the sender's and receiver's dumping files. Using a NS-2 based simulation network, one can test a new SVC video transmission algorithm, or evaluate the performance of SVC video transmission over a conventional network model (supported by NS-2). A simulated network can comprise of many relaying nodes. Since the SVC bit-stream comprises of multiple layers, enhancement layers can be discarded at the relaying nodes according to the simulation scripts.

- ***SVC Re-builder***: Being the heart of EvalSVC, the Re-builder will collect all data from sender's, receiver's dumping and video trace files, take both the SVC encoded bit-stream and the hinted file at the sender into account and reconstruct a possibly-corrupted output SVC bit-stream at the receiver. The SVC re-builder must understand SVC NALU headers in order to properly rebuild the corrupted SVC bit-stream. When encountering a missing packet, or a missing frame, the SVC re-builder has two options. It can truncate the SVC video frame or fill that frame with zero (or a default value). Other QoS measurements of the network such as end-to-end delay, jitter, loss rate, sender's and receiver's bit-rate will also be calculated.

- Error Generator: Normally, an optimal transmission condition can be obtained by using a direct connection between a sender and a receiver. We can use the Error Generator to modify the dump and trace files according to a pre-defined error distribution function.

- SVC Evaluator: This component will compare the bit-stream from the output of the SVC Re-builder with the original bit-stream from the sender. Objective and subjective quality evaluation (PSNR, MOS) of the SVC video transmission will be carried out at this component.

- Sender/Receiver nodes: Real or simulated nodes on the transmission network. They are the departure and destination of the video transmission.

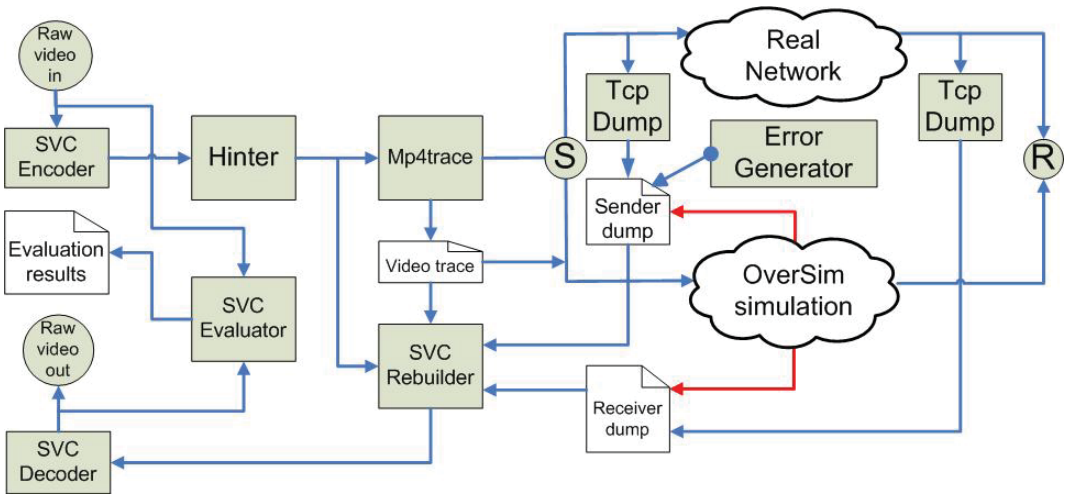


Figure 2.1: EvalSVC's diagram.

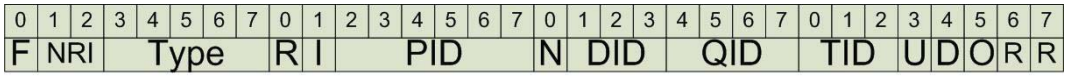


Figure 2.2: SVC NALU's header.

A sample evaluation session using EvalSVC starts with the raw video taken from a file or real-time captured by a camera. This raw video will then be encoded by the SVC encoder to form a SVC bit-stream. The SVC encoded bit-stream is fed into the Hinter to be packetized into RTP packets. A hint track will also be added to the original bit-stream. Mp4trace will send the hinted file (using streaming or camera mode) from the Sender node to the real/simulated network. A video trace file, a sender and a receiver dumping files will be generated. Using information from all of these files, and the original bit-stream, the SVC Re-builder will reconstruct the received bit-stream and feed it to the SVC Evaluator for generating the video transmission results. The reconstructed SVC video can also be delivered to the SVC decoder to get the output video play-out at the receiver side.

2.2.3 Evaluation metrics

2.2.3.1 Objective metric 1 - Peak signal-to-noise ratio

Peak signal-to-noise ratio (PSNR) is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. More specifically, here we use Y-PRNR to compare the PSNR of the luminance (Y) component of the videos. For objective measurement, EvalSVC uses the Peak Signal to Noise Ratio (PSNR) frame by frame. In YUV video, since the human's eyes are more sensible with the luminance component of the video

than with color components, EvalSVC calculates PSNR of the luminance component Y of source image S and destination image D.

$$Y - PSNR(s,d) = 20\log_{10}\left(\frac{V_{peak}}{MSE(s,d)}[dB]\right) \tag{2.1}$$

$$MSE(s,d) = \sqrt{\frac{1}{N_{col}N_{row}}\sum_{i=0}^{N_{col}}\sum_{j=0}^{N_{row}}[Y_S(n,i,j) - Y_D(n,i,j)]^2} \tag{2.2}$$

In which:

- $V_{peak} = 2^k - 1$
- k=number of bits per pixel (luminance component)

2.2.3.2 Objective metric 2 - Structural SIMilarity Index

Structural SIMilarity Index (SSIM) is a method for measuring the similarity between two images. It was designed to improve the traditional methods like Peak Signal-to-Noise Ratio (PSNR) and Mean Squared Error (MSE), which have proved to be inconsistent with human eye perception.

$$SSIM(x,y) = \frac{(2\mu_x\mu_x + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

In which:

- μ_x, μ_y are the mean values of x and y,
- σ_x^2, σ_y^2 are the variance values of x and y,
- σ_{xy} is the covariance of x and y,
- c_1, c_2 are two variables to stabilize the division with weak denominator;

2.2.3.3 Subjective metric

For subjective measurement, we use Mean Opinion Score (MOS) [56], which scales the human quality impression on the video from bad (0) to excellent (5). After having the PSNR measurements, the corresponding MOS scales can be found by using the conversion table 2.1.

2.3 Proposed EvalSVC's interfaces

2.3.1 Proposed interface to real networks

If the VCL is the interface between the encoder and the actual video frames, the Network Abstraction Layer (NAL) is the interface between that encoder and the actual network

protocol, which will be used to transmit the encoded bit-stream. The NAL encoder encapsulates the output slices of the VCL encoder into Network Abstraction Layer Units (NALU), which are suitable for transmission over packet networks or used in packet oriented multiplex environments [49]. In order to generate proper NAL units, we must pre-define the network protocol that we want to use to transmit the video bit-stream. H.264/AVC and SVC support encapsulating VCL slices into a number of network protocols (H.320, MPEG-2, and RTP...) [50] in which RTP is mostly used because of its popularity.
SVC extended the H.264/AVC standard by providing scalability. There are three main kinds of scalability that SVC can support: Temporal, spatial, quality (SNR).

2.3.2 Proposed interface to Overlay simulation platform

Oversim [58] is a simulation platform for overlay networks. In comparison to NS-2, it can provide better peer-to-peer and overlay simulation features. We can easily simulate application layer multicast algorithms (such as NICE, Narada...) with an almost unlimited number of peers within a multicasting group. Nowadays, more and more visual services (IPTV, video conferencing...) are being provided on multicast overlay networks. Our evaluation platform has an interface to the Oversim platform so that a scalable video bit-stream generated from our platform can be multicasted from a source node over the overlay simulated network generated by OverSim. Then, at each receiving peers within that multicast group, a possibly corrupted bit-stream will be reconstructed and compared with the original bit-stream. SSIM and PSNR measurements will be carried out at any peer or all peers of that multicast group when necessary. This feature is favourable for visual service designers and researchers of application layer multicast algorithms to verify and evaluate their proposals. EvalSVC makes use of the trace files of actual SVC bit-streams. Instead of sending the real video which has big sizes and often has copyrighted contents, trace files containing frame sizes and sending timescales will be used. We can make use of the available on-line scalable video coding trace library [59]. We can also generate a trace file from a specific SVC bit-stream by using the *mp4trace* block. According to that trace file, an application running on a randomly chosen source peer of the multicast group will generate the SVC traffic and transmit it through the simulated overlay. At the same time, it creates a sender's dumping file and store them

Table 2.1: Conversion between PSNR and MOS [57].

PSNR [dB]	MOS
> 37	5 (Excellent)
31 - 37	4 (Good)
25 - 31	3 (Fair)
20 - 25	2 (Poor)
< 25	1 (Bad)

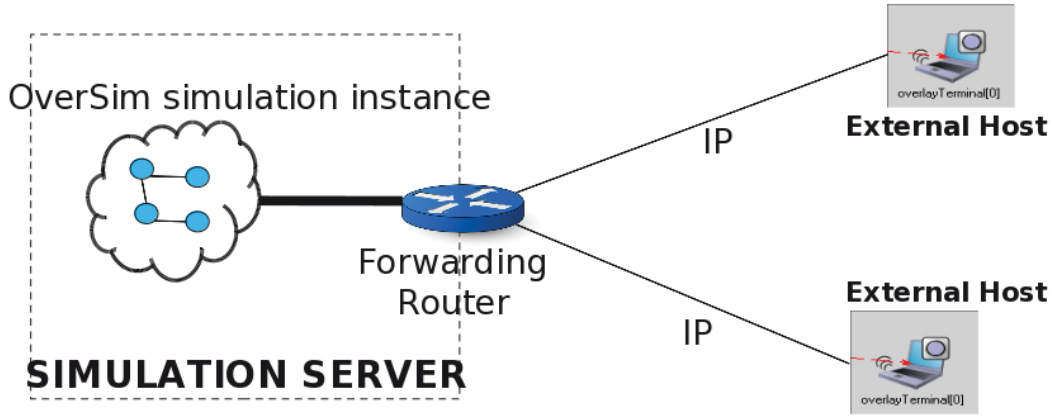


Figure 2.3: Hybrid network settings.

at the sender's side. The video packets are then transmitted on the multicast group to other peers. Each peer will generate a receiver dumping file and write an entry to that file whenever it receives a packet from the sender via the multicast group. After the video transmission session ends, receiver's dumping files are collected from all receiving peers. The information from the sender dumping/trace files, the original/hinted bit-streams and the receivers' dumping files at receiving peers, possibly corrupted bit-streams are reconstructed at each receiver. These files can be decoded using a common Scalable Video Decoder and then compared with the original raw video at the sender using common methods such as Y-PSNR and SSIM and MOS.

2.3.3 Hybrid network environment

Previously, EvalSVC either supports simulation network or real network but not two types of environment at the same time. There are many cases when we need to simulate a heterogeneous network in which several terminals are using different types of access networks to participate into an overlay network. Figure 2.3 demonstrates a hybrid network environment in which the evaluation of the SVC transmission can be done on a hybrid environment of simulated and real networks. While the simulation server applies the INET underlay Internet generated by GT-ITM, the external hosts apply a single-host underlay to connect to the simulation server via real forwarding routers. Using this kind of hybrid network environment, evaluations can be made when only a limited number of real terminal equipments are provided.

2.4 Performance comparison of AVC vs. different types of SVC end-to-end transmission on EvalSVC in a bottleneck network environment

Our first evaluation scenario is to compare the end-to-end transmission performance of different types of SVC scalability and AVC performances in bottle-neck environment. We try to find out the best SVC method which has the highest resistance against the bottleneck condition of the network since bottleneck is a very common condition on today's network [46]. The evaluation process and platform's architecture are demonstrated in Fig. 2.4. A sample evaluation session using EvalSVC starts with the raw video taken from a file or real-time captured by a camera. This raw video will then be encoded by the SVC encoder to form a SVC bit-stream. The SVC encoded bit-stream is fed into the Hinter to be packetized into RTP packets. A hint track will also be added to the original bit-stream. Mp4trace will send the hinted file (using streaming or camera mode) from the Sender node to the real/simulated network. A video trace file, a sender and a receiver dumping files will be generated. Using information from all of these files, and the original bit-stream, the SVC Re-builder will reconstruct the received bit-stream and feed it to the SVC Evaluator for generating the video transmission results. The reconstructed SVC video can also be delivered to the SVC decoder to get the output video play-out at the receiver side. For example, a cif-size raw file with 1065 frames is encoded using SNR SVC. The output bit-stream is sent via a real direct IP connection from a sender to a receiver. We manually generate errors by erasing entries at the sender's and receiver's dumping files. At the receiver, the received bit-stream is re-constructed by using the re-builder component of EvalSVC. Since the JSVM decoder cannot decode a corrupted bit-stream, we need to extract the uncorrupted base-layers out of the corrupted bit-stream for it to be decoded by the decoder. We can also use EvalSVC to evaluate the transmission of different kinds of SVC streams on a simulated network using NS-2.
In the first simulation scenario (Fig. 2.5), we try to find out if SVC transmission is really better than AVC transmission in the bottle neck condition of the network and if it is the case, which type of SVC scalability would be the best. To simulate the bottleneck condition, 3 nodes are built using NS-2: node 0 (the sender), node 1 (the relay), and node 2 (the receiver). The first link (link 1), connecting node 0 and node 1, has a bandwidth of 400 kbps, 1 ms delay. The second link (link 2), connecting node 1 and node 2 has a bandwidth of 100 Mbps, 1 ms delay. This network configuration will create a bottleneck on link 1. Firstly, a CIF-size AVC stream is sent from node 0 to node 2 via node 1. In the second and third simulations, a SNR SVC stream and a Spatial SVC stream (both CIF-size) are sent respectively via the same route from node 0 to node 2. We use EvalSVC to evaluate the end-to-end transmission performance of these 3 streams with all three different metrics (PSNR, SSIM, and MOS).
Figure 2.6 shows that, when bottom-neck occurs, SNR SVC has the best and AVC has the worst Y-PSNR performance. MOS grades of AVC, spatial SVC, and SNR SVC streams are 1.02, 4.07, and 5, respectively (for subjective measurement, Mean Opinion

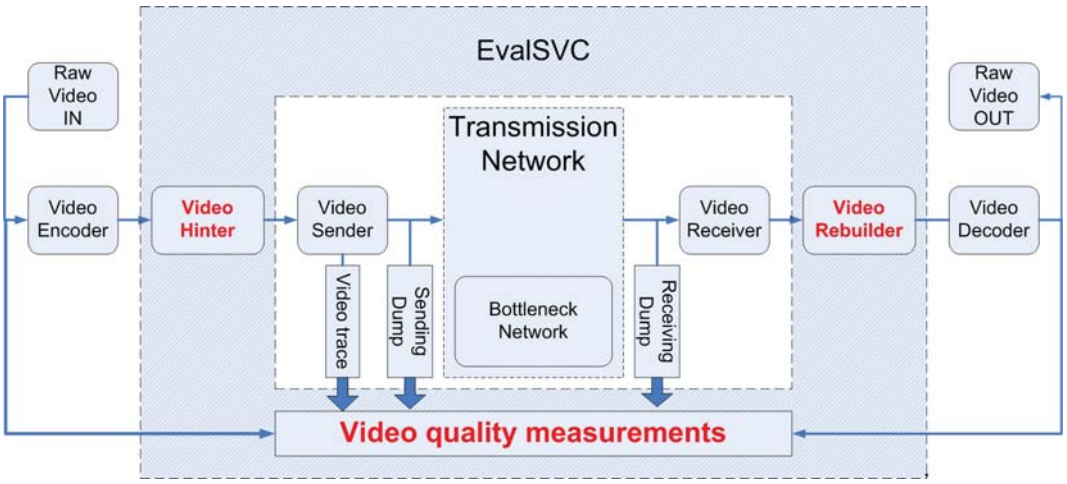


Figure 2.4: EvalSVC with the evaluation of the SVC transmission on bottleneck network.

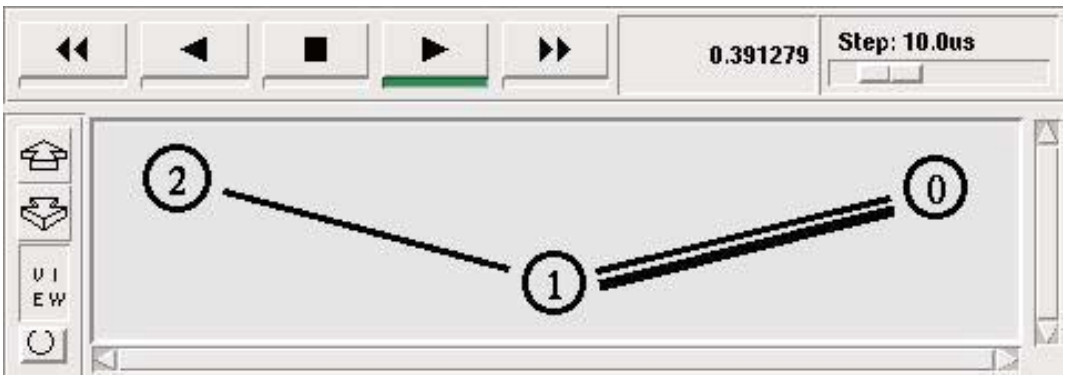


Figure 2.5: NS-2 based simulation diagram of video transmission over a bottleneck network.

Table 2.2: Simulation parameters of the SVC transmission over bottleneck network.

Parameters	**Values**
Purpose	Compare performance of AVC and different types of SVC on bottleneck condition
Encoding	• AVC, • SVC: Temporal, Spatial, SNR
Video size	CIF
Transmission network	Bottleneck condition
Network simulation tool	NS-2
Network condition	• Link 1: 400 kbps bandwidth, 1ms delay, • Link 2: 100 Mbps bandwidth, 1ms delay.
Video quality related measurements	• Objective: Y-PSNR, SSIM, • Subjective: MOS

Score (MOS) is used, which scales the human quality impression on the video from bad (0) to excellent (5)). There is also a convert table between PSNR and MOS values [56]. We can conclude that, AVC is very sensible to bottleneck, a single bottleneck in the transmission route can easily block the entire communication session. Meanwhile, all SVC streams can perform quite well (MOS > 4) under the bottleneck condition of the network, among those, SNR SVC has the best performance. Figure 2.7 shows the performance comparison of the three bit-streams using the structural similarity (SSIM) index. The SSIM measurement on the same bottle-neck simulated conditions gives us the similar result with Y-PSNR when compared SNR, spatial scalability and AVC. With a structural similarity, the SSIM metric can better reflect the video transmission quality for human visual system. As it can be analysed from Fig. 2.7, at the beginning, the quality of the AVC is good. Then, after the first 100 frames, the quality is suddenly dropped down due to transmission problems until the 500^{th} frame. During that period, the video quality is very bad and the received video content is absolutely not similar to the sending video content and therefore almost not recognisable by the users. The quality recovers by the 500^{th} frame and then continues to be acceptable until frame 750, and then it becomes very bad again until the end. We cannot see this video quality evolution

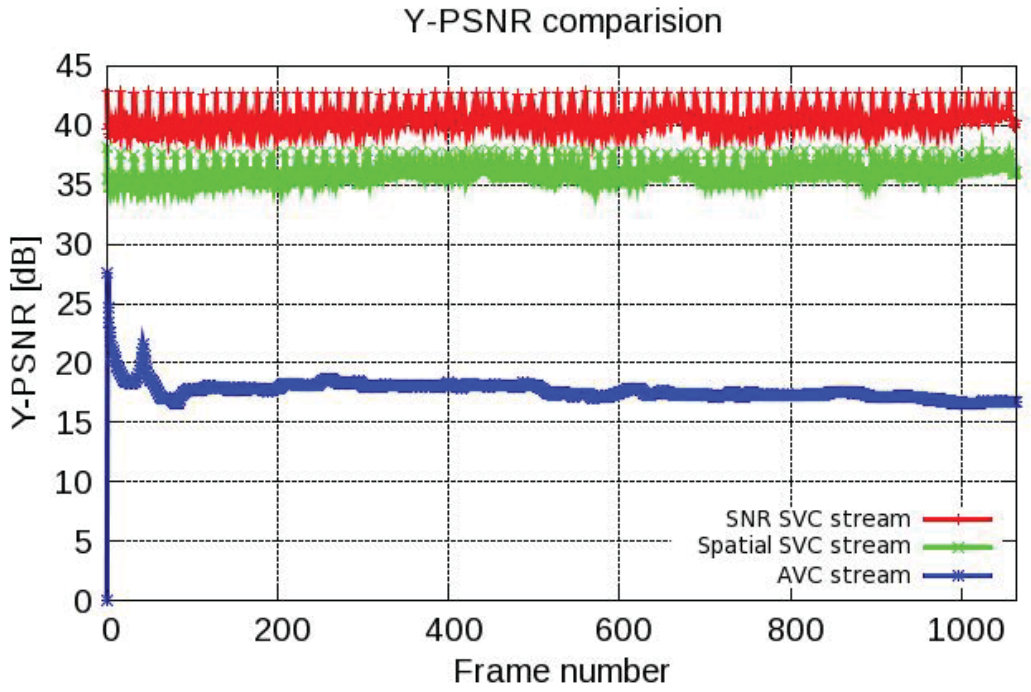


Figure 2.6: Y-PSNR comparision among AVC, SNR SVC and Spatial SVC streams.

if we only use the an average metric like PSNR in Fig. 2.6. This big variation in video quality will be perceived very badly by the users. For example, a part of the movie or an important discussion of the video communication session can be missed. From the end user Quality of Experience (QoE) point of view, a change from a high video quality to a very bad video quality brings a very bad user experience. We can see from Fig. 2.7 with the SSIM metric that AVC transmission creates very bad QoE because of its high variation in video transmission quality. Different types of SVC scalability transmission is better not only because it has a higher PSNR performance but also because it has a lower variation in SSIM.

Figure 2.8 shows the frame loss flags of different types of video encoding methods. A value of 1 (one) means the corresponding frame is lost while a value of 0 (zero) means the frame is successfully received. According to the frame loss flag of the AVC transmission, we can find that, the AVC video starts to loose its frames from frame number 100. The frames from 200 to about 500 are heavily lost. The density of the frame loss flags are lower from frames 600 to about 750. Frame numbers from 800 to 900 suffer a big loss. Since the SSIM value is more sensible with the loss of the B frames, we can recognize its variance corresponding to the density of the AVC's frame loss flag. Meanwhile, the Y-PSNR result in Fig. 2.6 shows that the Y-PSNR is not sensible with the frame loss. From the results in Fig. 2.6 and Fig. 2.7, we can conclude that SVC has a better resistance than AVC in the bottleneck conditions of the network and SNR SVC has the

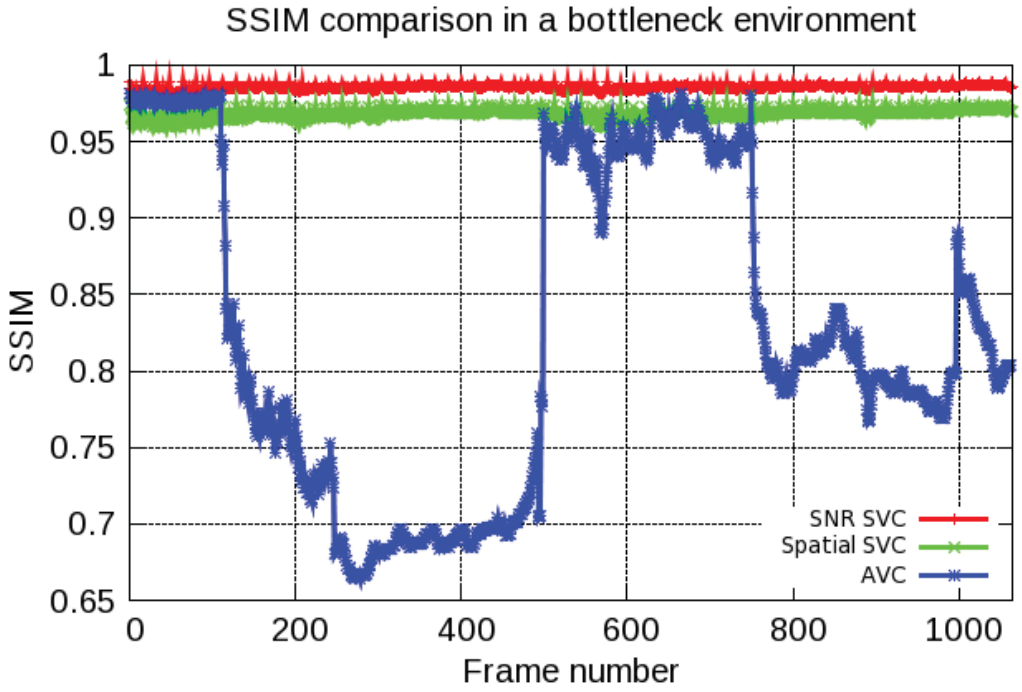


Figure 2.7: Structural similarity index comparison among SNR, spatial SVC and AVC.

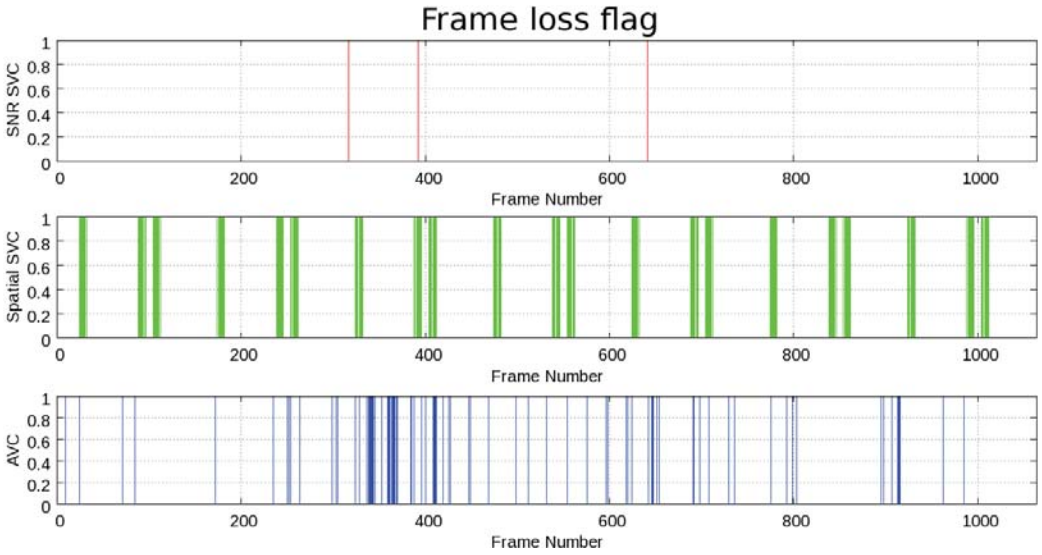


Figure 2.8: Frame loss flag among AVC, SNR SVC and Spatial SVC transmission on bottleneck condition.

best end-to-end transmission performance. Since bottleneck condition is very common in overlay network, we recommend using SVC, and more specifically SNR SVC, to build video services on overlay network.

2.5 Performance comparison of AVC vs. different types of SVC end-to-end transmission on EvalSVC over an overlay network environment

We evaluate here the end-to-end transmission performance of AVC vs. different types of SVC scalability contents on the overlay network. We use a popular overlay network (NICE) to transmit the video contents in this evaluation scenario [60]. In order to evaluate the transmission performance of different types of SVC on the overlay environment, the evaluation process and platform architecture are demonstrated in Fig. 2.9. EvalSVC makes use of the trace files of actual SVC bit-streams. Instead of sending the real video which has big size and often has copyrighted contents, trace files containing frame sizes and sending time-scales will be used. We can make use of the on-line available scalable video coding trace library [59]. We can also generate a trace file from a specific SVC bit-stream by using the *mp4trace* block. According to that trace file, an application running on a randomly chosen source peer of the multicast group will generate the SVC traffic and transmit it through the simulated overlay. At the same time, it creates a sender's dumping file and store them at the sender's side. The video packets are then transmitted on the multicast group to other peers. Each peer will generate a receiver dumping file and write an entry to that file whenever it receives a packet from the sender via the multicast group. After the video transmission session ends, receiver's dumping files are collected from all receiving peers. The information from the sender dumping/trace files, the original/hinted bit-streams and the receivers' dumping files at receiving peers, possibly corrupted bit-streams are reconstructed at each receiver. These files can be decoded using a common Scalable Video Decoder and then compared with the original raw video at the sender using common methods such as Y-PSNR and SSIM.
Figure 2.10 shows the Y-PSNR measurement of SVC and AVC video transmission over the OverSim interface. We can see that, regarding the Y-PSNR on an ALM environment, SNR-SVC has the best performance followed by spatial-SVC, and temporal-SVC (it should be noted that the temporal-SVC bit-stream has the smallest number of frames simply because many B frames have been dropped for scalability). AVC still owns the worst performance. Regarding the SSIM measurement among the same set of video over the ALM environment [7], Fig. 2.11 show that SNR-SVC, and spatial-SVC still outperform temporal-SVC and AVC.
The conclusion to withdraw from these results is that SVC transmission performs better than AVC transmission on overlay network and SNR SVC has the best quality when transmitted on overlay network. Therefore, SVC and more specifically SVC's SNR scalability will be used in the next parts of the research as the main video encoding scheme.

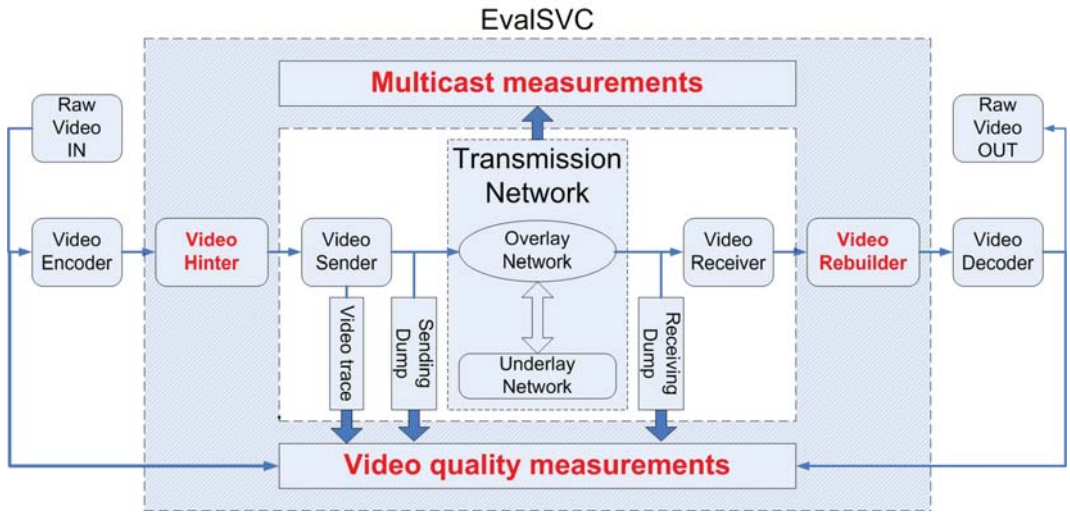


Figure 2.9: EvalSVC and the performance evaluation of different types of SVC transmission on overlay network.

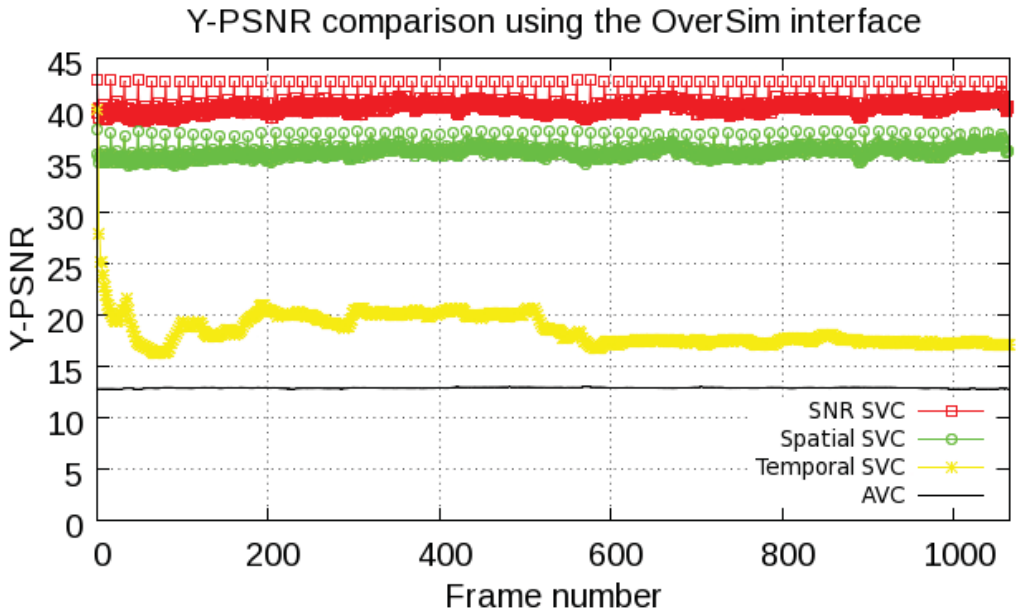


Figure 2.10: Y-PSNR comparison among SNR, spatial, temporal SVC and AVC transmission performance over the OverSim interface.

Table 2.3: Simulation parameters of the different types of SVC transmission on overlay network constructed from using conventional cost function.

Parameters	**Values**
Purpose	Compare video quality related performance of SVC transmissions on overlay network constructed by using a popular cost function
Encoding	SVC: Temporal, SNR, and Spatial
Video size	CIF
Transmission network	Overlay network
Service	Application Layer Multicast of SVC video
Network simulation tool	Oversim
Number of peer	1-1024 peers
Underlay network	Internet topology generated by GT-ITM
Cost functions	NICE's popular cost function
Video quality related measurements	• Y-PSNR • SSIM

2.6 Conclusion

In this chapter, we have introduced EvalSVC, a new evaluation platform for end-to-end transmission of Scalable Video Coding contents. The first purpose of this work is to fill the gap between the design, evaluation and implementation processes of variable visual services based on Scalable Video Coding. We have proposed new interfaces between the EvalSVC platform and the bottleneck and overlay network environments. Output measurement results are also provided in three different metrics: MOS, PSNR and SSIM to better reflect the SVC-contents end-to-end transmission performance.
The first contribution of this chapter is to provide all necessary information of the EvalSVC platform's architecture to help the user to handle the platform more easily. We leave the platform as an open-source evaluation tool[9] on line for the industrial and research community. Therefore, it is easy for them to handle the source codes, apply and integrate into their systems, develop and extend the functionalities if necessary.
The second contribution is to conduct end-to-end transmission performance evaluation of SVC contents. Two main evaluation scenarios have been conducted. The first scenario is the end-to-end transmission performance comparison of AVC vs. different types of SVC scalability contents on EvalSVC in a bottleneck network environment. The second scenario is the end-to-end transmission performance comparison of AVC vs. different types of SVC scalability contents on EvalSVC over an overlay network environment.

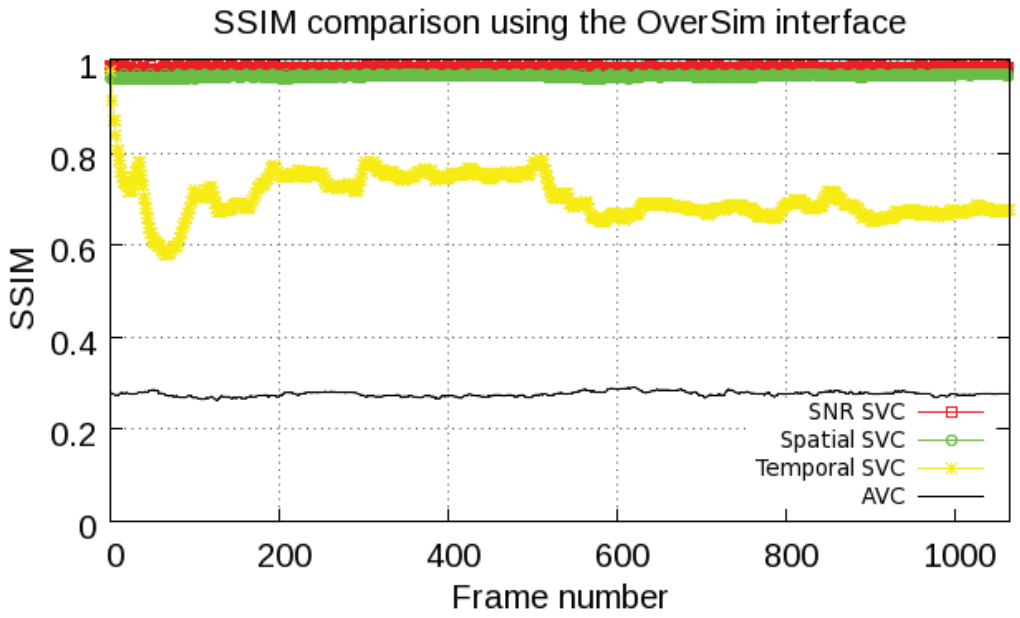


Figure 2.11: SSIM comparison among SNR, spatial, temporal SVC and AVC transmission performance over the OverSim interface.

Using our newly developed EvalSVC platform and the end-to-end transmission evaluation scenarios, we found that, SVC transmission is better than AVC transmission both in bottleneck conditions and on overlay network. Among different types of SVC's scalability, through end-to-end transmission evaluation results, we found that, SNR scalability has the highest PSNR, MOS as well as SSIM performance while transmitted both in bottleneck conditions and on the overlay network.
Through these transmission performance evaluations, we can conclude that, indeed SVC achieves its objectives of improving video transmission over realistic network conditions. After having found the suitable SVC scalability for a multimedia service, we will study the multicast mechanism for multimedia content over an overlay network in the following part of this PhD thesis.

CHAPTER 3

Application-Network Cross Layer Multi-variable Cost Function for multimedia delivery over Convergent Networks

Contents

3.1 Introduction . . . 33
3.2 Conventional approaches . . . 34
3.3 Proposed cross-layer multi-variable cost function . . . 36
3.3.1 Problem formulation . . . 36
3.3.2 Lemma 1: Bandwidth-type cost function . . . 36
3.3.3 Lemma 2: Delay-type cost function . . . 38
3.3.4 Derivation of the multi-variable cost function . . . 38
3.3.5 N-variable cost function with $N \geq 3$ . . . 42
3.4 Theoretical analysis of the newly proposed cross-layer multi-variable cost function . . . 42
3.5 Proposed cross-layer cost function performance on multimedia delivery platforms . . . 43
3.5.1 Video multicast service on overlay network . . . 45
3.5.2 Video conferencing service in convergent underlay network environment . . . 48
3.6 Conclusion . . . 51

3.1 Introduction

Application Layer Multicast (ALM) algorithms are either similar or conceptually based on Network Layer Multicast's cost functions. In this chapter, a new application-network cross layer multi-variable cost function is proposed. It optimises the variable requirements and available resources from both the application and the network layers. It can dynamically update the available resources required for reaching a particular node on the

ALM's media distribution tree. Mathematical derivation and theoretical analysis have been provided for the newly proposed cost function so that it can be applied in more general cases of different contexts. An evaluation platform of an overlay network built over a convergent underlay network comprised of a simulated Internet topology and a real 4G mobile WiMAX IEEE802.16e wireless network is constructed. The performance of the newly proposed cross-layer multi-variable cost function is evaluated and compared with conventional cost functions in Scalable Video Coding-based services such as video multicast and video conferencing services using the evaluation platform. Intensive evaluation results have shown that the newly proposed cross-layer multi-variable cost function outperforms the conventional cost function and improves the performance of multimedia services on dynamic convergent network environments.

3.2 Conventional approaches

The concept of cost function originally came from the theoretical Steiner tree problem in geometry. Nevertheless, it became a research topic in communication for solving problems on routing of multipoint connections [61]. Since routing is the main responsibility of the network layer in the OSI model, it is intuitable that conventional cost functions are mainly built based on the network layer's routing approach.
We only consider application layer cost function in this thesis, so network layer cost functions and routings such as the ones proposed in [62] are out of the scope of this research. Since the idea of Application Layer routings is new, the following state-of-the-art cost functions are so far the only related works that we can find.
Conventional cost functions are either empirical or heuristic [63][64]. Among all available cost functions for ALM routing that we have found, none of them has a mathematical derivation nor a clear citation. This does not mean those cost functions are not good since many of them are heuristically practical, but it is very difficult for designers to understand the reason why they should select a particular cost function and what are its exact reactions over network's conditions and application's requirements. Therefore a cost function with theoretical analysis and mathematical background is highly demanded for the ease of its applications. In most of the ALM routing algorithms, the state of the network, on which the routing algorithm is presented, readily associates some costs with each link. Thus they do not address how the link cost function should be defined so as to efficiently distribute allocated resources over the network [65]. Also in [65], several kinds of cost functions have been investigated.

$$LinkCost = \frac{RsvBw + ReqBw}{LinkCap} \tag{3.1}$$

In which:

- RsvBw: The amount of bandwidth currently in use by existing connections,
- ReqBw: The amount of bandwidth requested by the newly arriving group of participants,

- LinkCap: Total bandwidth of the link.

The main idea of using the cost function Equ.3.1 is to choose a tree that is least-loaded and at the same time, to minimize the total amount of bandwidth to be consumed by the new connection. A variant which accounts for the length of the path (in number of hops) is also considered in [65] as follows:

$$LinkCost = 1 + \frac{RsvBw}{LinkCap} \tag{3.2}$$

The main idea of using the cost function of Equ.3.2 (a variation of Equ.3.1) is that hop count, a static link metric, plays a dominant role in the link cost.

$$LinkCost = \frac{1}{LinkCap - (RsvBw + ReqBw)} \tag{3.3}$$

In Equ.3.3, the cost increases exponentially with the utilization of the link. According to [65], Equ.3.3 appears to be more attractive than Equ.3.1 and Equ.3.2 because it can better distribute the load over the network by avoiding the use of highly loaded links, and thus links don't likely become saturated for future connections. Furthermore, it gives preference to shorter paths (with less number of link) as long as links are not heavily loaded.
In [66], the cost function Equ.3.4 is used because it has many desirable practical characteristics. It decreases with the delay, it is convex, it assigns infinitely high cost when the required delay guarantee approaches zero, and a fixed minimal link usage cost C^l, even if no guarantee is required. The constant θ determines how fast the cost grows for low delays and the constant A^l is used as a scaling constant.

$$C^l(d) = \frac{A^l}{d^\theta} \tag{3.4}$$

In [67], the cost function on each link is calculated by Equ.3.5

$$c_i(x) = \begin{cases} \dfrac{\kappa_i}{x - \kappa_i} & , \text{ if } x > \kappa_i \\ \infty & , \text{ if } x \leq \kappa_i \end{cases} \tag{3.5}$$

In which:

- κ_i: Minimal delay that can be guaranteed on link i when utilizing all of its available resources,
- x_i: Requested delay of the new connection.

Although delay is used as a sample, any other QoS parameter such as bandwidth, jitter, packet loss can be used. However, the heuristics function Equ.3.5 is only single-variable, therefore it cannot consider other parameters simultaneously.

Another cost function Equ.3.6 considering several QoS parameters has been used in [68], again, without any mathematical proof.

$$C(\beta, \delta, \psi) = \delta + \frac{K_1}{\beta} + K_2.e^{\frac{K_3}{\psi}} \tag{3.6}$$

In which:

- β: Residual bandwidth,
- δ: Residual buffer space,
- ψ: Estimated delay bound.

The scaling factor K_i allows us to modulate the relationship between β, δ, ψ even further, although it is still unclear how bandwidth, buffer, and delay units could be added exactly together [68],[69].
In [70], a single-variable cost function has been derived and its performance has been simulated under real conditions. The results have shown that, the single-variable cost function can only build a good multicast tree in certain conditions. A multi-variable cost function is highly demanded to build a more reliable multicast tree. From the best of our knowledge, conventional cost functions are mainly applied in network-layer routings. Our work has been one of the pioneers who have made full investigations on application layer multicast routing considering both network conditions and application's requirements.

3.3 Proposed cross-layer multi-variable cost function

Assuming that we have an overlay network with application peers and end-to-end links, in order to form a tree for data delivery, we need to know the costs of all those end-to-end links. These costs must be calculated by a cost function. To take into account several QoS parameters simultaneously, the cost function must be a cross-layer multi-variable function. On each end-to-end link, we have to consider variable requirements from applications running on the ALM-based overlay. For example, an application can be a scalable video service with different video coding layers or it can be a multimedia flux comprising of video, audio, text, data sub-streams, each has different bandwidth and delay requirements. Those requirements are varied frequently by the application. We have to also consider the maximum available resources of the underlay network. For example, if an end-to-end link is built upon 3 physical links, each has its own available bandwidth and delay. Then the maximum available bandwidth of the end-to-end link equals to the minimum available bandwidth (bottleneck) of all 3 physical links, the minimum guaranteed delay of the end-to-end link equals to the sum of all guaranteed delays on the 3 physical links.
QoS parameters can be either bandwidth-type (meaning that the requested bandwidth is always smaller than or equal to the maximum available bandwidth) or delay-type (meaning that the requested delay is always greater than or equal to the minimum available delay).

3.3.1 Problem formulation

Problem: *Find a multi-variable cost function which can simultaneously consider varied bandwidth and delay requests from the application and maximum guaranteed resources from the underlay network. The cost function must be able to assign increasingly higher costs for nearly-saturated end-to-end links to prevent congestion.*

3.3.2 Lemma 1: Bandwidth-type cost function

Assume we have on the end-to-end *link i*: A total available bandwidth of κ_w, and a requested bandwidth of x_w, we must find the bandwidth-type cost function: $f(x_w)$. Since κ_w is the maximum available bandwidth when using all available resources on *link i*, so $0 \leq x_w \leq \kappa_w$. With time, according to the application's requirements, x_w may be varied by an amount of Δx_w causing the cost to have the current value of $f(x_w + \Delta x_w)$, so this current value of the cost function depends on:

- The previous cost, $f(x_w)$: The cost value at any time depends on its previous cost before there is any variation in the application's requirement,
- The increment of cost which is proportional to:
  - The previous cost, $f(x_w)$: If the previous cost is high, the trend is that the increment of the cost should be proportionally high to prevent the multicast from saturation. On the other hand, if the previous cost value is not very much high, the increment of the cost can be reasonably low to provide enough space for the cost to raise,
  - The ratio between the increment of requested bandwidth and the total requested bandwidth: $\frac{\Delta x_w}{x_w + \Delta x_w}$,
- The decrement of cost which is proportional to:
  - The ratio between the decrement of the remaining available bandwidth and the maximum available bandwidth $\frac{(\kappa_w - x_w - \Delta x_w)}{\kappa_w}$: It is obvious that if this ratio is high, so is the decrement of cost.

Finally, we have:

$$f(x_w + \Delta x_w) = f(x_w).\left[1 + \frac{\dfrac{\Delta x_w}{x_w + \Delta x_w}}{\dfrac{(\kappa_w - x_w - \Delta x_w)}{\kappa_w}}\right] \tag{3.7}$$

From Equ.3.7 we have:

$$\Leftrightarrow f'(x_w) = f(x_w).\frac{\kappa_w}{x_w(\kappa_w - x_w)} \tag{3.8}$$

Replacing $f(x_w)$ by y and $f'(x_w)$ by $\frac{dy}{dx_w}$; from Equ.3.8 we have an ordinary differential equation:

$$\frac{dy}{dx_w} = y\frac{\kappa_w}{x_w(\kappa_w - x_w)} \tag{3.9}$$

Solve the ordinary differential equation Equ.3.9, we find the bandwidth-type cost function:

$$\begin{aligned}
&\frac{dy}{y} = \frac{\kappa_w}{x_w(\kappa_w - x_w)}dx_w \Leftrightarrow \int\frac{dy}{y} = \int\frac{\kappa_w}{x_w(\kappa_w - x_w)}dx_w \\
&\Leftrightarrow \int\frac{x_w + (\kappa_w - x_w)}{x_w(\kappa_w - x_w)}dx_w = -\int\frac{d(\kappa_w - x_w)}{\kappa_w - x_w} + \int\frac{dx_w}{x_w} \\
&\Leftrightarrow \ln(y) = \ln(x_w) - \ln(\kappa_w - x_w) + c \Leftrightarrow y = \frac{\Phi.x_w}{(\kappa_w - x_w)}
\end{aligned} \tag{3.10}$$

3.3.3 Lemma 2: Delay-type cost function

We can see that, the required delay parameter (x_d) has a reversed characteristic against the required bandwidth parameter (x_w). So by replacing $\dot{x}_d = \frac{1}{x_d}$, $\dot{\kappa}_d = \frac{1}{\kappa_d}$, and $d\dot{x}_d = d(\frac{1}{x_d}) = -\frac{dx_d}{{x_d}^2}$ into Equ.3.9 we have:

$$\frac{dy}{d\dot{x}_d} = y\frac{\dot{\kappa}_d}{\dot{x}_d(\dot{\kappa}_d - \dot{x}_d)} \Leftrightarrow -\frac{dy}{\frac{dx_d}{{x_d}^2}} = y\frac{\frac{1}{\kappa_d}}{\frac{1}{x_d}(\frac{1}{\kappa_d} - \frac{1}{x_d})} \Leftrightarrow \frac{dy}{dx_d} = y\frac{1}{\kappa_d - x_d} \tag{3.11}$$

Equation 3.11 is the ordinary differential equation to derive the delay-type cost function. From Equ.3.11, we have:

$$\begin{aligned}
&\frac{dy}{y} = \frac{dx_d}{\kappa_d - x_d} \Leftrightarrow \int\frac{dy}{y} = -\int\frac{d(x_d - \kappa_d)}{x_d - \kappa_d} \\
&\Leftrightarrow \ln(y) = -\ln(x_d - \kappa_d) + \ln(c) \Leftrightarrow y = \frac{c}{x_d - \kappa_d} \Leftrightarrow y = \frac{\Psi.\kappa_d}{x_d - \kappa_d}
\end{aligned} \tag{3.12}$$

3.3.4 Derivation of the multi-variable cost function

We now try to derive the **bandwidth-delay cost function** $u(x_w, x_d)$ considering two independent QoS parameters: bandwidth (x_w) and delay (x_d) at the same time. From Equ.3.9 and 3.11, we have:

$$\frac{x_w(\kappa_w - x_w)}{\kappa_w}u_{x_w} + (\kappa_d - x_d)u_{x_d} = u \tag{3.13}$$

In which $u_{x_w} = \frac{\partial u}{\partial x_w}$, and $u_{x_d} = \frac{\partial u}{\partial x_d}$.
Equation 3.13 is a *quasi linear first order partial differential equation*, we will solve it to

obtain our bandwidth-delay cost function.
Let $x_w = x_w(s)$, $x_d = x_d(s)$, $u = u(x_w(s), x_d(s))$, then:

$$\frac{\partial x_w}{\partial s}.u_{x_w} + \frac{\partial x_d}{\partial s}.u_{x_d} = \frac{\partial u}{\partial s} \tag{3.14}$$

Compare Equ.3.13 and 3.14, we have:

$$\begin{cases} \dfrac{\partial x_w}{\partial s} = \dfrac{x_w(\kappa_w - x_w)}{\kappa_w} \\ \dfrac{\partial x_d}{\partial s} = \kappa_d - x_d \\ \dfrac{\partial u}{\partial s} = u \end{cases} \tag{3.15}$$

A *constant of integration* is obtained by eliminating s from two or more equations and integrating out. Such integration generates an arbitrary integration constant, which may be viewed as a function of all the variables, but it is constant with respect to s. Let $\phi(x_w, x_d, u)$ be a constant of integration, since it is constant with respect to s, we can write:

$$\begin{aligned} \frac{d\phi}{ds} = 0 \Leftrightarrow \frac{\partial \phi}{\partial x_w}.\frac{\partial x_w}{\partial s} + \frac{\partial \phi}{\partial x_d}.\frac{\partial x_d}{\partial s} + \frac{\partial \phi}{\partial u}.\frac{\partial u}{ds} = 0 \\ \Leftrightarrow \frac{\partial \phi}{\partial x_w}.\frac{x_w(\kappa_w - x_w)}{\kappa_w} + \frac{\partial \phi}{\partial x_d}.(\kappa_d - x_d) + \frac{\partial \phi}{\partial u}.u = 0 \end{aligned} \tag{3.16}$$

Equation 3.16 is the *orthogonality property* of the vector (x_w, x_d), we can use it to check whether ϕ has been obtained correctly.
In order to solve Equ.3.13 we have to find two constants of integration from Equ.3.15.

3.3.4.1 Finding the first constant of integration:

From Equ.3.15, we have:

$$\frac{\dfrac{dx_w}{ds}}{\dfrac{du}{ds}} = \frac{\dfrac{x_w(\kappa_w - x_w)}{\kappa_w}}{u} \Leftrightarrow \frac{du}{u} = \frac{\kappa_w.dx_w}{x_w(\kappa_w - x_w)}$$

Since Equ.3.17 and Equ.3.9 have an identical form, we can use Lemma 1 to achieve the first constant of integration Equ.3.17 :

$$u = \frac{\Phi.x_w}{(\kappa_w - x_w)} \Leftrightarrow \Phi = \frac{(\kappa_w - x_w)u}{x_w} \tag{3.17}$$

We now check the orthogonality property of the first constant of integration Equ.3.17 by confirming Equ.3.16:

$$\begin{aligned}
&\frac{\partial\Phi}{\partial x_w}.\frac{x_w(\kappa_w - x_w)}{\kappa_w} + \frac{\partial\Phi}{\partial x_d}.(\kappa_d - x_d) + \frac{\partial\Phi}{\partial u}u \\
&= \frac{\partial(\frac{(\kappa_w - x_w)u}{x_w})}{\partial x_w}.\frac{x_w(\kappa_w - x_w)}{\kappa_w} + \frac{\partial(\frac{(\kappa_w - x_w)u}{x_w})}{\partial x_d}(\kappa_d - x_d) + \frac{\partial(\frac{(\kappa_w - x_w)u}{x_w})}{\partial u}u \\
&= \frac{-\kappa_w.u}{x_w^2}.\frac{x_w(\kappa_w - x_w)}{\kappa_w} + 0.(\kappa_d - x_d) + \frac{(\kappa_w - x_w)}{x_w}.u = 0 \quad (3.18)
\end{aligned}$$

By Equ.3.18 and Equ.3.16, we can confirm that the first constant of integration has been correctly found.□

3.3.4.2 Finding the second constant of integration:

Similarly, using Equ.3.15 and Equ.3.12, we can find the second constant of integration having the form of:

$$\Psi = \frac{(x_d - \kappa_d)u}{\kappa_d} \quad (3.19)$$

We now check the orthogonality property of the first constant of integration Equ.3.19 by confirming Equ.3.16:

$$\begin{aligned}
&\frac{\partial\Psi}{\partial x_w}.\frac{x_w(\kappa_w - x_w)}{\kappa_w} + \frac{\partial\Psi}{\partial x_d}.(\kappa_d - x_d) + \frac{\partial\Psi}{\partial u}u \\
&= \frac{\partial(\frac{(x_d - \kappa_d)u}{\kappa_d})}{\partial x_w}.\frac{x_w(\kappa_w - x_w)}{\kappa_w} + \frac{\partial(\frac{(x_d - \kappa_d)u}{\kappa_d})}{\partial x_d}(\kappa_d - x_d) + \frac{\partial(\frac{(x_d - \kappa_d)u}{\kappa_d})}{\partial u}u \\
&= 0.\frac{x_w(\kappa_w - x_w)}{\kappa_w} + \frac{u}{\kappa_d}.(\kappa_d - x_d) + \frac{(x_d - \kappa_d)}{\kappa_d}.u = 0 \quad (3.20)
\end{aligned}$$

By Equ.3.20 and Equ.3.16, we can confirm that the second constant of integration has been correctly found.□

3.3.4.3 The general solution:

The equation $\Phi(x_w, x_d, u) =$ constant, describes a relationship among x_w, x_d, u such as shown in Equ.3.15. Notice that if $\Phi(x_w, x_d, u)$ is a constant, then $\mathcal{G}(\Phi)$ is also a constant ($\mathcal{G}(.)$ is any arbitrary function). Similarly, if $\Psi(x_w, x_d, u)$ is a constant, then $\mathcal{H}(\Psi)$ is also a constant ($\mathcal{H}(.)$ is any arbitrary function). We can set these two constants equal so that:

$$\mathcal{G}(\Phi) = \mathcal{H}(\Psi) \quad (3.21)$$

This provides a more general expression in x_w, x_d, u that solves the partial differential equation Equ.3.13. The two arbitrary functions in Equ.3.21 may be merged into one by letting $\mathcal{F}(.) = \mathcal{G}^1(\mathcal{H}(.))$, then:

$$\Phi = \mathcal{F}(\Psi) \quad (3.22)$$

Equ.3.22 is the general solution of the partial differential equation Equ.3.13.

3.3.4.4 Checking the general solution:

From Equ.3.17, Equ.3.19, Equ.3.22, we have:

$$\frac{(\kappa_w - x_w)u}{x_w} = \mathcal{F}\left(\frac{(x_d - \kappa_d)u}{\kappa_d}\right) \tag{3.23}$$

We will now check whether Equ.3.23 is indeed a solution, and that the function $\mathcal{F}$ can be completely general. Let u be expressed as a function of x_w and x_d, where x_w and x_d are still independent.
We take the total derivative of Equ.3.23 to x_w and solve for u_{x_w}. Denoting $\mathcal{F}\left(\frac{(x_d-\kappa_d)u}{\kappa_d}\right) = \mathcal{F}(x_d, u)$, we have:

$$\begin{aligned} &\frac{\kappa_w - x_w}{x_w}.u_{x_w} - \frac{\kappa_w}{{x_w}^2}.u = \frac{x_d - \kappa_d}{\kappa_d}.u_{x_w}.\mathcal{F}'(x_d, u) \\ \Leftrightarrow\ & u_{x_w} = \frac{\kappa_w.\kappa_d.u}{x_w\left[(\kappa_w - x_w)\kappa_d - x_w(x_d - \kappa_d)\mathcal{F}'(x_d, u)\right]} \end{aligned} \tag{3.24}$$

Similarly, taking the total derivative of Equ.3.23 to x_d and solve for u_{x_d}, we have:

$$\begin{aligned} &\frac{(\kappa_w - x_w)u_{x_d}}{x_w} = \mathcal{F}'(x_d, u).\left[\frac{u + (x_d - \kappa_d)u_{x_d}}{\kappa_d}\right] \\ \Leftrightarrow\ & u_{x_d} = \frac{\mathcal{F}'(x_d, u).u.x_w}{(\kappa_w - x_w)\kappa_d - \mathcal{F}'(x_d, u).(x_d - \kappa_d).x_w} \end{aligned} \tag{3.25}$$

Replacing Equ.3.24 and Equ.3.25 into the left-hand side of Equ.3.13 we have:

$$\begin{aligned} &\frac{x_w(\kappa_w - x_w)}{\kappa_w}.\frac{\kappa_w.\kappa_d.u}{x_w.\left[(\kappa_w - x_w)\kappa_d - x_w(x_d - \kappa_d).\mathcal{F}'(x_d, u)\right]} + \\ &\quad ... + (\kappa_d - x_d).\frac{\mathcal{F}'(x_d, u).u.x_w}{(\kappa_w - x_w)\kappa_d - \mathcal{F}'(x_d, u)(x_d - \kappa_d).x_w} \\ &\quad = \frac{(\kappa_w - x_w).\kappa_d - (x_d - \kappa_d).\mathcal{F}'(x_d, u).x_w}{(\kappa_w - x_w)\kappa_d - x_w(x_d - \kappa_d).\mathcal{F}'(x_d, u)}.u = u \end{aligned}$$

We can obtain the right-hand side of Equ.3.13, so Equ.3.23 is exactly the general solution of the partial differential equation Equ.3.13.

3.3.4.5 Fitting boundary conditions to the general solution:

Equation Equ.3.23 provides us a general solution comprising of a family of arbitrary functions. We need to fix to a certain bandwidth-delay cost function by assigning boundary conditions to this general solution. From the natural characteristics of two independent QoS parameters: *requested bandwidth* (x_w) and *requested delay* (x_d), and their partial

cost functions Equ.3.10 and Equ.3.12, we have these boundary conditions:

$$\begin{cases} \dfrac{x_w}{\kappa_w - x_w} = t^3 \\ \dfrac{\kappa_d}{x_d - \kappa_d} = t \\ u = t^2 \end{cases} \tag{3.26}$$

Replacing Equ.3.26 into Equ.3.23 we have:

$$\mathcal{F}(\frac{t^2}{t}) = \frac{t^2}{t^3} \Leftrightarrow \mathcal{F}(t) = \frac{1}{t} \tag{3.27}$$

From Equ.3.27 and Equ.3.23, we have:

$$\mathcal{F}\left(\frac{(x_d - \kappa_d).u}{\kappa_d}\right) = \frac{\kappa_d}{(x_d - \kappa_d).u} = \frac{(\kappa_w - x_w)u}{x_w} \tag{3.28}$$

The specific solution of Equ.3.13 is therefore:

$$u(x_w, x_d) = \sqrt{\frac{x_w}{\kappa_w - x_w}.\frac{\kappa_d}{x_d - \kappa_d}}$$

3.3.5 N-variable cost function with $N \geq 3$

Naturally, one may concern about the existence of a cross-layer multi-variable cost function when the number of variable is more than two. More specifically, what happens if we want to consider not only bandwidth and delay requirements from the application but also the other popular requirements such as jitter, packet-loss rate etc.. We may have full list of them. Thus, how to obtain the cost function in such case is a right question to ask.
Recursively from Equ.3.29, we can see that, the specific multi-variable cost function equals to the average multiplication of all partial cost functions $f_i(x_i)$:

$$u(x_1, x_2, ..., x_n) = \sqrt[n]{\prod_{i=1}^{n} f_i(x_i)} \tag{3.29}$$

In which:

- $x_1, x_2, ..., x_n$: List of application requirements (e.g. bandwidth, delay, jitter, packet-loss rate etc.),
- $f_i(x_i)$: Partial cost functions.

In order to obtain the N-variable cost function with $N \geq 3$, we first need to define the list of application requirements $(x_1, x_2, ..., x_n)$. Afterwards, we must find a partial cost function for each of these application requirements. The processes described in subsections 3.3.2 and 3.3.3 can be followed to obtain the partial cost functions.

In general, we can build a cost function for as many variables as possible given separated partial cost functions (e.g. jitter-type and packet-loss rate type cost functions). However, while a cross-layer multi-variable cost function can consider many QoS parameters at the same time, it should be noticed that the cross-layer multi-variable cost function does not always give a better result than the single-variable cost function. For example, the cost function with bandwidth, delay, and packet-loss can build a better multicast tree if many peers are using wireless access network with a high packet loss rate to join the multicast tree but when most of the peers are using a wired access network with a low packet loss rate, then that three-variable cost function may build a worse multicast tree than the two-variable cost function of only bandwidth and delay. Therefore, a N-variable cost function with $N \geq 3$ should be designed and applied with care.

3.4 Theoretical analysis of the newly proposed cross-layer multi-variable cost function

Conventional cost functions can be divided into two groups. The first group comprises of single-variable cost functions given by Equ.3.1, Equ.3.2, Equ.3.3, Equ.3.4, and Equ.3.5. The other group comprises of multi-variable cost functions such as Equ.3.6. They are described in section 3.3.

Figure 3.1(a) shows the value of a bandwidth-type single-variable cost function versus the requested bandwidth. Since $x_w < \kappa_w$, we only consider the left domain of the graph. We can see that, when the requested bandwidth approaches κ_w, the link cost increases rapidly to infinity. Figure 3.1(b) shows the value of a delay-type single-variable cost function versus the requested delay. Since $x_d > \kappa_d$, we only consider the right domain of the graph. We can see that, when the requested delay approaches κ_d, the link cost increases rapidly to infinity. Nevertheless, two single-variable functions failed to consider other important QoS parameters simultaneously. Figure 3.2 shows the value of the newly proposed cross-layer multi-variable cost function. In this graph, only two QoS parameters (requested bandwidth and delay) are shown for demonstrative purposes. We only use the right part of the graph where $x_w < \kappa_w$ and $x_d > \kappa_d$. The main difference is that, the link cost will only increase rapidly when both requested bandwidth x_w and delay x_d excess the maximum resources κ_w and κ_d. When only a single parameter excesses its limited resource, a high cost will be assigned, however, it should not be high enough to block the entire link since we still have resources to assign for the other parameter. For example, when x_d excesses κ_d, the cost increases to a high value, however, if we still have much bandwidth to assign, then the increased link cost will not be sufficient to block the entire link. The link cost will only increase rapidly if both the requested bandwidth and delay simultaneously excess their available resources (which are the total resources that the link can provide).

Compared to the heuristic multi-variable cost function Equ.3.6, our new cross-layer multi-variable cost function has many advantages. Firstly, it has considered the requested QoS parameters from the application instead of only the residual resource from

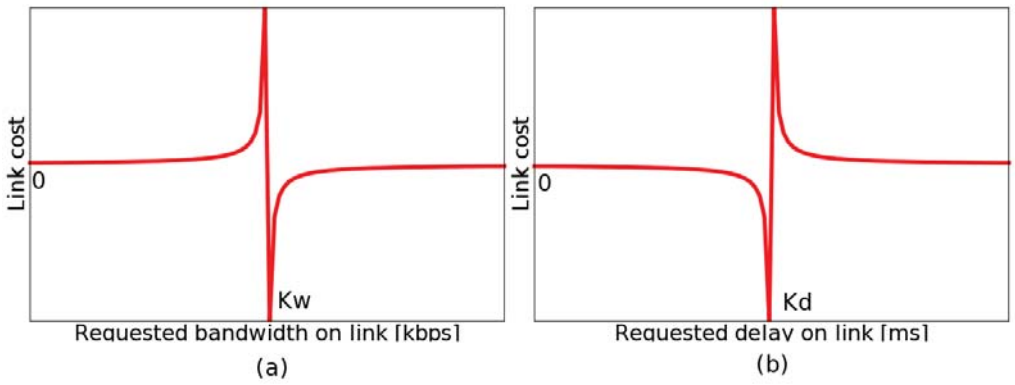


Figure 3.1: Bandwidth-type (a) and delay-type (b) cost functions according to Equ.3.10 and Equ.3.12.

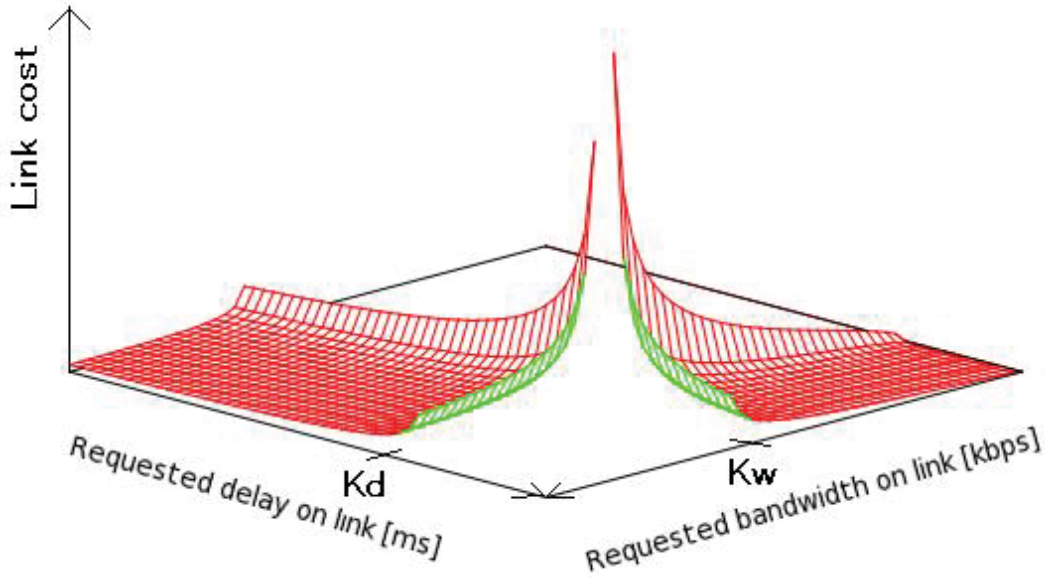


Figure 3.2: Cross-layer multi-variable cost function according to Equ.3.29.

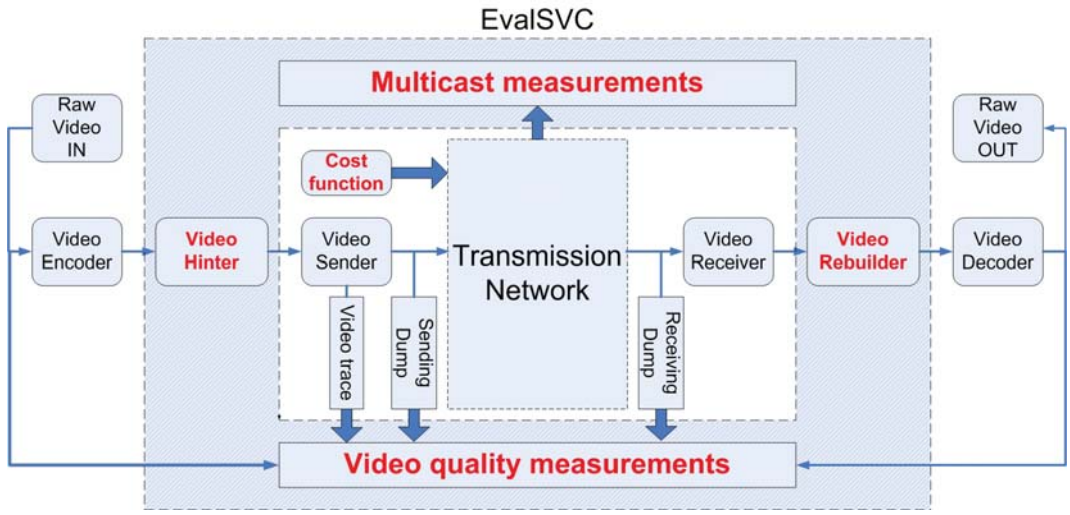


Figure 3.3: EvalSVC: An evaluation platform of SVC-based services.

the underlay network. This is a major advantage since application's requirements are usually varied. Secondly, although the scaling factors K_i allows us to assign different weights for different parameters, it is not clear how to assign values for these factors in practice, therefore, there will be implementation problems. Thirdly, it is really unclear how we can add all bandwidth, buffer and delay together without any unit impairment. From the above analysis, we can conclude that our new cross-layer multi-variable cost function owns better characteristics than conventional cost functions.

3.5 Proposed cross-layer cost function performance on multimedia delivery platforms

After having the new cross-layer multi-variable cost function, it is necessary to evaluate its performance. Since it has been proved that SVC-content can resist better in the convergent environment of the overlay network, an evaluation platform of the newly proposed cross-layer multi-variable cost function with SVC content is highly required. EvalSVC is such a new evaluation platform for SVC content transmission [8]. The evaluation platform is shown in Fig.3.3. In this platform, the performance of the new cost function can be evaluated by comparing the video-related and multicast-related measurements of video transmissions on different transmission network's conditions.
We will use the four popular metrics for evaluation: (i) average link stress, (ii) link stress, (iii) average end-to-end delay, and (iv) end-to-end delay to compare the performance between the new cost function and the conventional NICE's distance function. The reason for choosing these metrics is because they are usually used to compare the performance of Application Layer Multicast systems. If only one link is considered, the average values of average link stress and the average end-to-end delay becomes the exact values of link

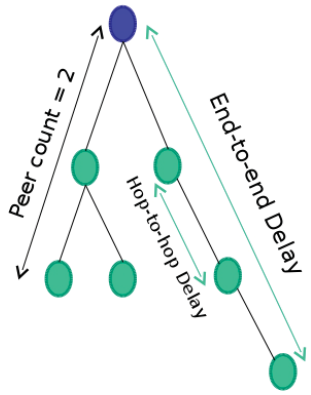


Figure 3.4: End to end delay.

stress and end-to-end delay.
Average link stress is defined in terms of the mean value of identical packets due to overlay forwarding, carried over a physical access link. This metric is equal to 1 for IP multicast. The lower the average link stress, the better the performance.

$$\begin{cases} \overline{stress(s_{id}, p_{id})} = \dfrac{1}{N} \displaystyle\sum_{i=0}^{N-1} stress(s_{id}, p_{id})_i \\ \overline{stress} = \dfrac{1}{N} \displaystyle\sum_{s_{id}=0}^{N-1} \left(\dfrac{1}{P_{sid}} \displaystyle\sum_{p_{id}=0}^{P_{sid}-1} \overline{stress(s_{id}, p_{id})} \right) \end{cases} \tag{3.30}$$

In which:

- p_{id}: Packet with an identity of id,
- s_{id}: Source with an identity of id,
- $stress(s_{id}, p_{id})_i$: Stress of link i caused by packet p_{id} being sent from the source s_{id},
- N: Total number of overlay nodes participating in the simulation platform,
- $\overline{stress(s_{id}, p_{id})}$: Average value of the link stress $stress(s_{id}, p_{id})_i$,
- P_{sid}: Total number of packets sent from s_{id},
- $\overline{stress}$: Average link stress on the entire overlay network.

Link stress is the exact value when only one link (real mobile WiMAX wireless link) is involved. This value can be found by replacing $N = 1$ into Equ.3.30.
Average end-to-end delay is defined as the average value of the end-to-end delay on

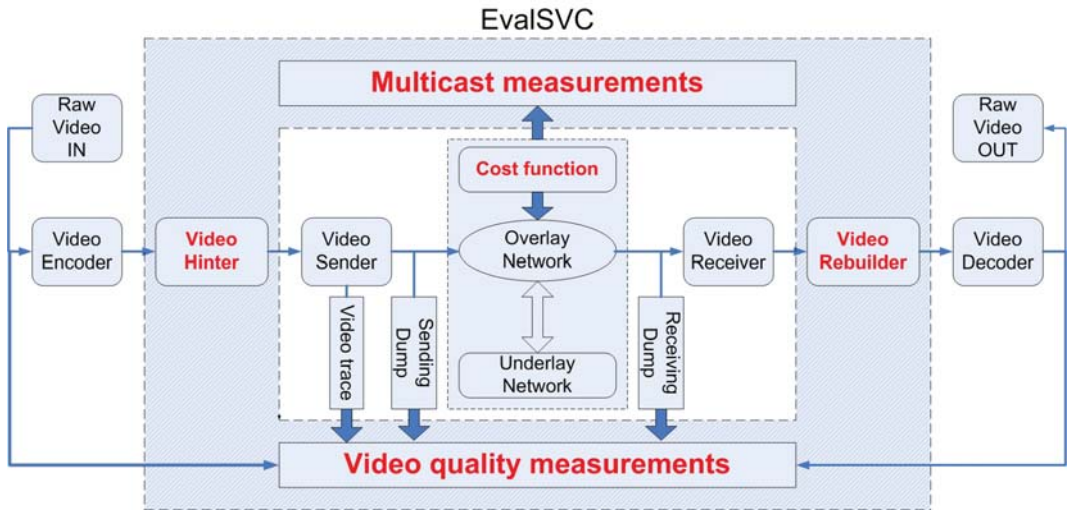


Figure 3.5: Performance evaluation platform of SVC-based multicast service on an overlay network constructed from our newly proposed cross-layer multi-variable cost function.

the entire overlay network as demonstrated on Fig.3.4.

$$\begin{cases} \overline{peercount} = \dfrac{1}{P}\displaystyle\sum_{sid=0}^{N-1}\sum_{pid=0}^{P_{sid-1}}\overline{peercount(sid,pid))} \\ \overline{e2edelay} = \overline{peercount}.\overline{hop2hopdelay} \end{cases} \tag{3.31}$$

In which:

- $\overline{peercount}$: Average number of peers that a packet must travel from its source node to its destination,
- p_{id}: Packet with an identity of id,
- s_{id}: Source with an identity of id,
- $\overline{peercount(sid,pid))}$: Average number of a particular packet with a source identity of s_{id} and a packet identity of p_{id},
- $\overline{hop2hopdelay}$: Average delay between 2 hops,
- $\overline{peercount}$: Average number of peers that a particular packet must travel from a sending node to a receiving node,
- $\overline{e2edelay}$: Average end-to-end delay,

End-to-end delay is the exact value when only one link (real mobile WiMAX wireless link) is involved. This value can be found by replacing $N = 1$ into Equ.3.31.

3.5.1 Video multicast service on overlay network

To compare the performance of the new cost function with a popular cost function, the evaluation platform is set up as described in Fig.3.5. We set up an OverSim [58] simulation scenario based on NICE. The main goal of the simulation is to show that the newly proposed cost function can perform better than the popular NICE's distance function in terms of the average link stress and the average end-to-end delay. In this research, we only choose to compare our newly proposed cost function with NICE's distance function, but the method and the platform can be applied to compare it with any other cost function. The simulation plan will build an overlay of a varied number of peers (e.g., varied group sizes of 16, 32, 64, 128, 256, 512, and 1024) running on an underlay network topology of the Internet generated by GT-ITM [71]. Each topology was a two-level hierarchical transit-stub topology, containing 1250 nodes and about 6000 physical links [72]. Each physical link will have random values of delay, bandwidth, and PER (Packet Error Ratio). We will use the simulation plan described in [73] for performance comparison and evaluation purposes.

Table 3.1: Simulation parameters of the SVC transmission on overlay network constructed from the cross-layer multi-variable and conventional cost function.

Parameters	**Values**
Purpose	Evaluation of the new cross-layer multi-variable cost function for Video Multicast service
Encoding	SNR SVC
Video size	CIF
Transmission network	Overlay network
Service	Application Layer Multicast of SVC video
Network simulation tool	Oversim
Number of peer	1-1024 peers
Underlay network	Internet topology generated by GT-ITM
Cost functions	• New cross-layer multi-variable cost function, • NICE's popular cost function
Overlay measurements	Average link stress, average end-to-end delay

NICE only uses a delay-type cost function to build and to maintain its ALM tree (with a clustering, layering structure). By sending and receiving periodical heartbeat messages containing delays between nodes within a cluster, peers will decide whether it should elect a new cluster's leader. Changing cluster's leaders provokes changing and rebuilding the entire NICE tree. In its original paper [74], authors of NICE implemented the

delay-type cost function simply by using an end-to-end delay parameter. We now want to apply our new cost function obtained from Equ.3.29.
Figure 3.6 shows a sample ALM data delivery. We assign the value of κ_w on the end-to-end *link A-B* by selecting the minimum available bandwidth among all available bandwidths of physical links connecting *peer A* to *peer B* (e.g. links A-1, 1-2, B-2). Similarly, we get the value of κ_w on all other end-to-end links (links A-B, B-C, and A-D). The required bandwidth x_w on all end-to-end links has the same value equalling to the required bandwidth of the source. The minimum guaranteed end-to-end delay κ_d is equal to the sum of all physical links' delays. Depending on the application, we have a delay limit from the sending peer to receiving peers. For example, in Fig.3.6 (b), peer A delivers data to all other peers simultaneously with the same delay requirement, so the end-to-end required delay for delivering data to the farthest peer (peer D) should be smaller than the delay limit. On other relaying end-to-end links (e.g. links A-B, B-C), the required delays are partitioned proportionally to their minimum guaranteed end-to-end delay κ_d, respectively. We run an application on overlay peers. The application continuously changes values of its required bandwidth (x_w) and delay (x_d). Costs of all end-to-end links will be calculated and NICE will use them instead of the conventional delay cost to run their algorithm on. We will compare performances in two cases mainly by using two metrics: average link stress and average end-to-end delay defined previously [60]. The average link stress metric is defined by the mean value of identical packets sent by a protocol over each underlay link. To calculate the average link stress of the network, instead of standing on each link and counting identical packets, we let the nodes (peers/routers) count the link stress of all their links, and then take a half of the total sum. The reason for doing so is because in OverSim, it is easier to control nodes than links, meanwhile any physical link is always formed just by 2 nodes. The average link stretch is the ratio of average path length of the members of a protocol to the average path length of the members in the multi-unicast protocol. In our implementation, we just concentrate on the numerator: the average path length (the mean value of actual hops) that a data packet must go through from source to destination. For each packet received at an overlay peer, we will take its Time-To-Live information which is actually the hop-count value that it has to go through. Note that we just need to count the path length of packets routed by the ALM protocol, so we take the calculation at the overlay layer, not at the underlay layer. All simulation evaluation parameters are in Table 3.1.

Figure 3.7 shows that the newly proposed cost function when applied by NICE can reduce the average link stress to a smaller value than the original NICE's distance function. Figure 3.8 shows that the average end-to-end delay when applying the new cost function is much smaller than the old distance function especially when the group size increases. Even when the number of participants is 1024, the average end-to-end delay of the new cost function is just about 79 ms which is still smaller than the limitation value of 150 ms recommended by ITU-T for real-time communication services [75]. From the results we can see that, the new cost function can avoid multiple replication of packets on access

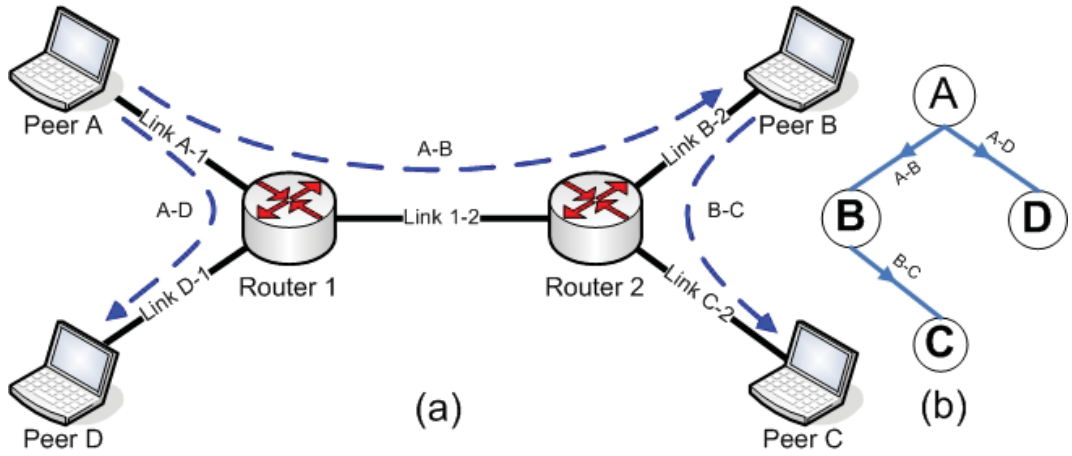


Figure 3.6: A sample ALM data delivery tree, (a) Overlay and underlay diagram, (b) Tree diagram.

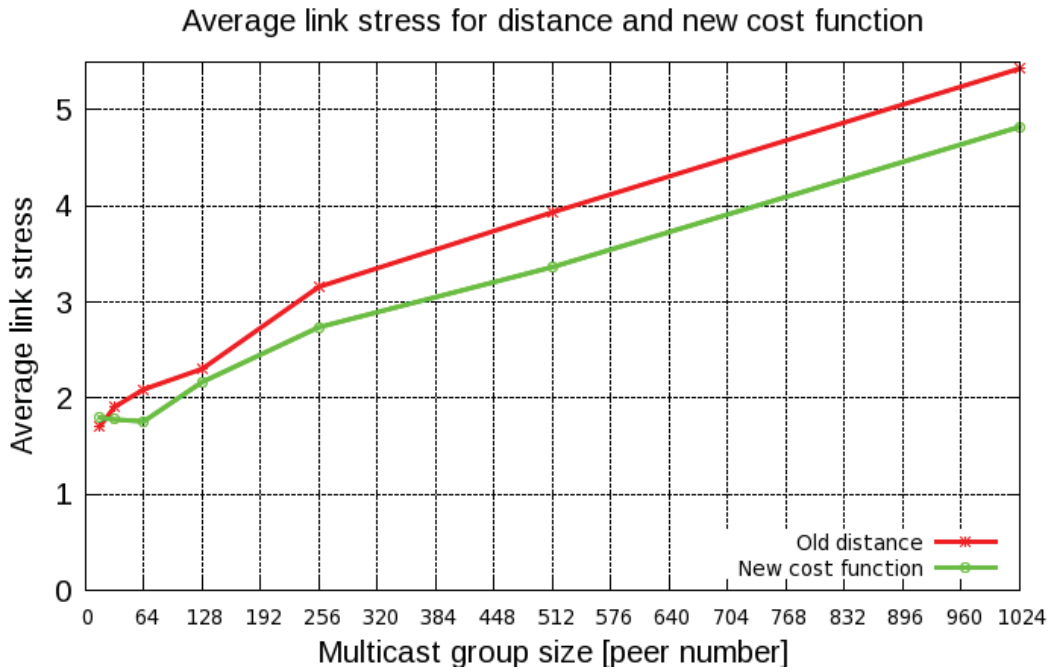


Figure 3.7: Average link stress comparison for the NICE data-plan using the old and new cost functions. Transmitting data is obtained from a real SVC multicast session.

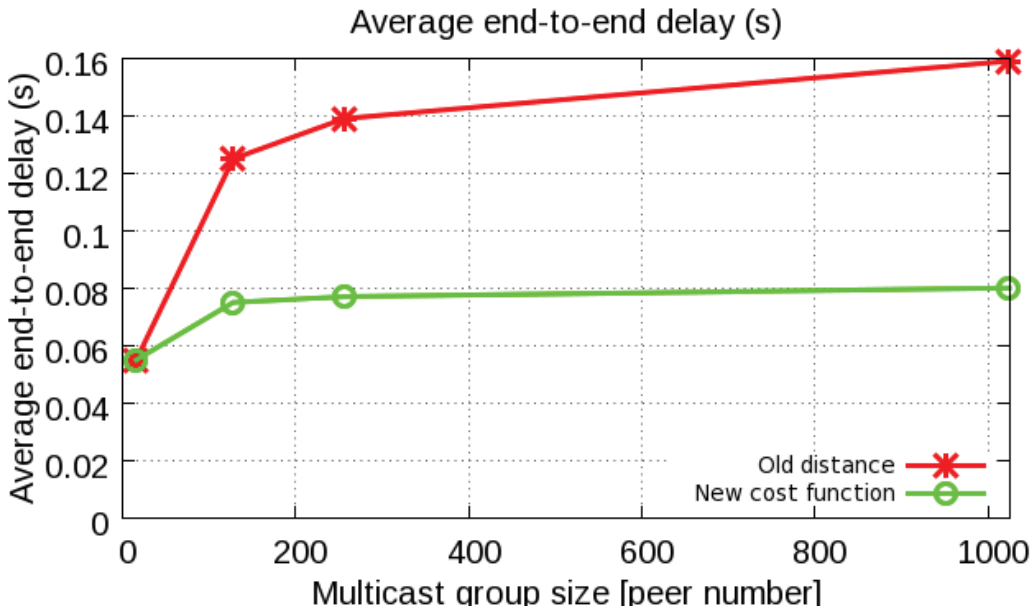


Figure 3.8: Average end-to-end delay comparison for the NICE data-plan using the old and new cost functions. Transmitting data is obtained from a real SVC multicast session.

links and therefore reduce the average link stress. Even though a packet may have to go through more physical hops in order to reach its destination, the new cost function can still guarantee a smaller average end-to-end delay than the conventional distance function. It should be noticed from Fig.3.8 that, when the group size is large, the new cost function can give out more routes for NICE to build its media distribution tree resulting in a much better average end-to-end delay than the conventional cost function.

3.5.2 Video conferencing service in convergent underlay network environment

To evaluate the performance of the newly proposed cost function in the real network conditions, we implement a test-bed based on both the Oversim-based simulation platform and a real mobile WiMAX network with two real WiMAX mobile terminals. The parameters of the test-bed are explained in Table 3.2. The convergent network is set up as illustrated in Fig.3.9. The real WiMAX mobile terminals connect to the simulated platform by using the real mobile WiMAX access network and core network provided by the French National Pole de Competitivite System@tic POSEIDON project [76]. The Oversim-based simulation platform is the same as the one from the previous simulation scenario. The mobile WiMAX access network comprises of an Acatel-Lucent extended Base Station (ALU xBS: 9710 C-WBS). The core network comprises of the Operations and Maintenance Centre (OMC), the Wireless Area Controller (WAC), the AAA (Authentication, Authorization, and Accounting), the HA (Home Agent), the DHCP (Dy-

Table 3.2: Simulation parameters of the SVC video conferencing service on the overlay network built over a convergent underlay network of simulated Internet topology and real mobile WiMAX network.

Parameters	**Values**
Purpose	Evaluation of the newly proposed cross-layer multi-variable cost function for Scalable Video Conferencing service on convergent network (simulated Internet + real WiMAX).
Video encoding	SNR SVC
Video size	CIF
Multicast	Overlay network
Transmission network	Simulated Internet topology and mobile WiMAX network
Number of WiMAX mobile terminals	2
Service	Application Layer Multicast for scalable video conferencing service
Network simulation tool	Oversim
Number of peer	1-1024 peers
Underlay network	Internet topology generated by GT-ITM
Cost functions	• New cross-layer multi-variable cost function, • NICE's popular cost function
Overlay measurements	• Average link stress, • Average end-to-end delay

namic Host Configuration Protocol), FTP and video servers. The first type of WiMAX mobile terminal is an Alcatel-Lucent 9799 PCMCIA card. The second type of WiMAX mobile terminal is a Sequans USB card. The IEEE 802.16e 2005 release technology (also called the "Mobile WiMAX" release) is used here.
Figure 3.10 illustrates the integration between the Oversim-based simulation platform and the real mobile WiMAX access network. This simulation scenario emulates a video conferencing service built on top of the ALM network. The participants can be divided into two groups. The first group comprises of simulated peers participating to the ALM

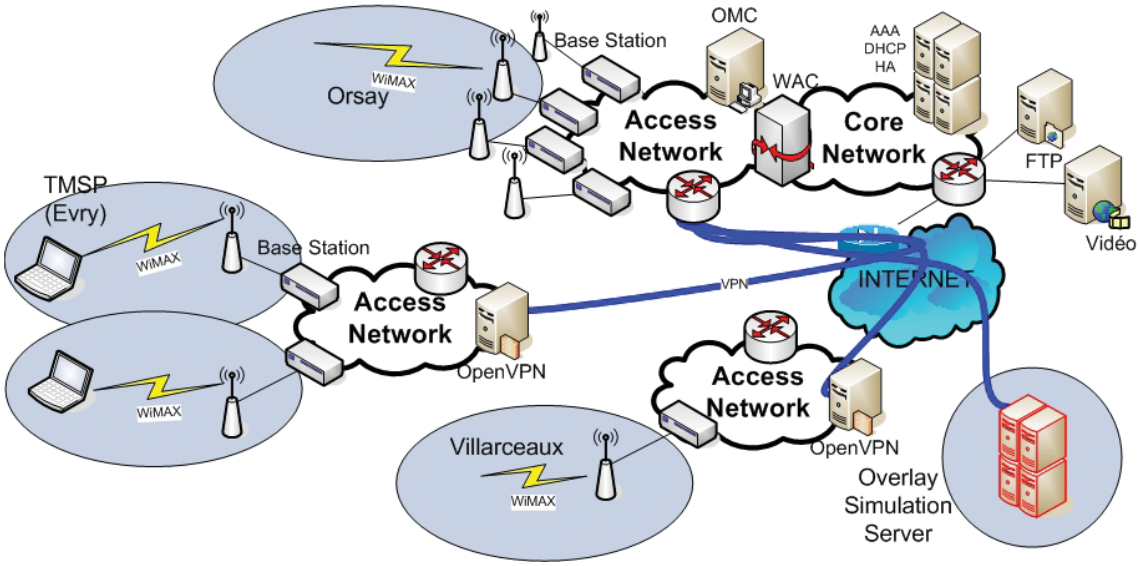


Figure 3.9: Convergent underlay network composed of real POSEIDON's Alcatel-Lucent mobile WiMAX network and an overlay simulation platform [76].

group from the INET [77] underlay network (a simulation topology of the Internet). We use 1 or 2 peer(s) participating into the ALM group from the mobile WiMAX network using the OMNET++ single host underlay [78]. A tunnelling interface is set up to connect between the main ALM group and the external WiMAX mobile peer(s).
Figure 3.11 shows the performance evaluation platform of SVC transmission on the overlay network built over a convergent underlay network of a simulated Internet topology and a real mobile WiMAX network. The Application Layer Multicast tree is constructed using the newly proposed cross-layer multi-variable cost function.
In Figures 3.12 and 3.13, the link stress, end-to-end delay and their average values are calculated at the mobile WiMAX access link and within the overlay network. In these two figures, the "Distance WiMAX" and "Cost WiMAX" labels show the performance of the conventional NICE's distance function and our newly proposed cost function on the mobile WiMAX access links, respectively. The "Distance-average" and "Cost-average" labels show the performance of the conventional NICE's distance function and our newly proposed cost function in average of all the links in the overlay network, respectively. In this case, the average values of both link stress and end-to-end delay are made over the entire overlay network including the simulated Internet topology and the mobile WiMAX underlay network.
Figures 3.12 and 3.13 show that, on the real WiMAX mobile link, our newly proposed cross-layer multi-variable cost function outperforms the conventional distance function in both the link stress and the end-to-end delay. On the overlay network, our newly proposed cross-layer multi-variable cost function also outperforms the conventional distance

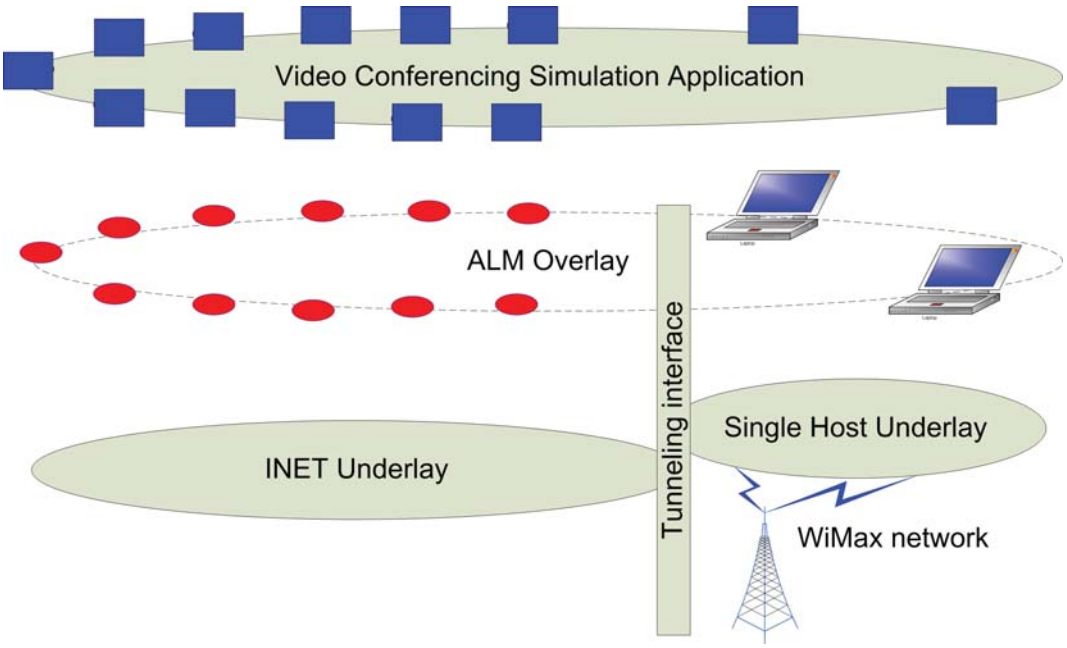


Figure 3.10: Integration between the Oversim-based simulation platform and the real mobile WiMAX access network.

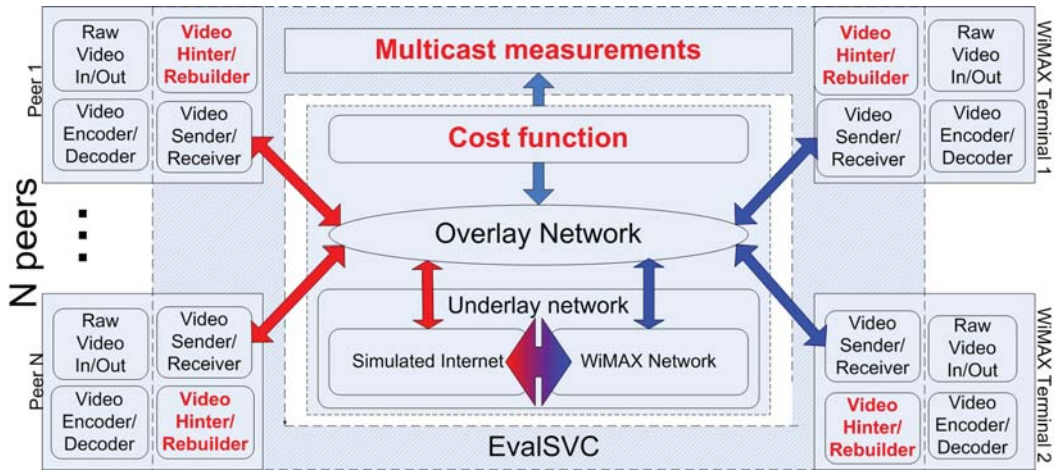


Figure 3.11: Performance evaluation platform of SVC-based conferencing services on the overlay network built over a convergent underlay network comprised of a simulated Internet topology and a real mobile WiMAX network. The Application Layer Multicast tree is constructed using the newly proposed cross-layer multi-variable cost function.

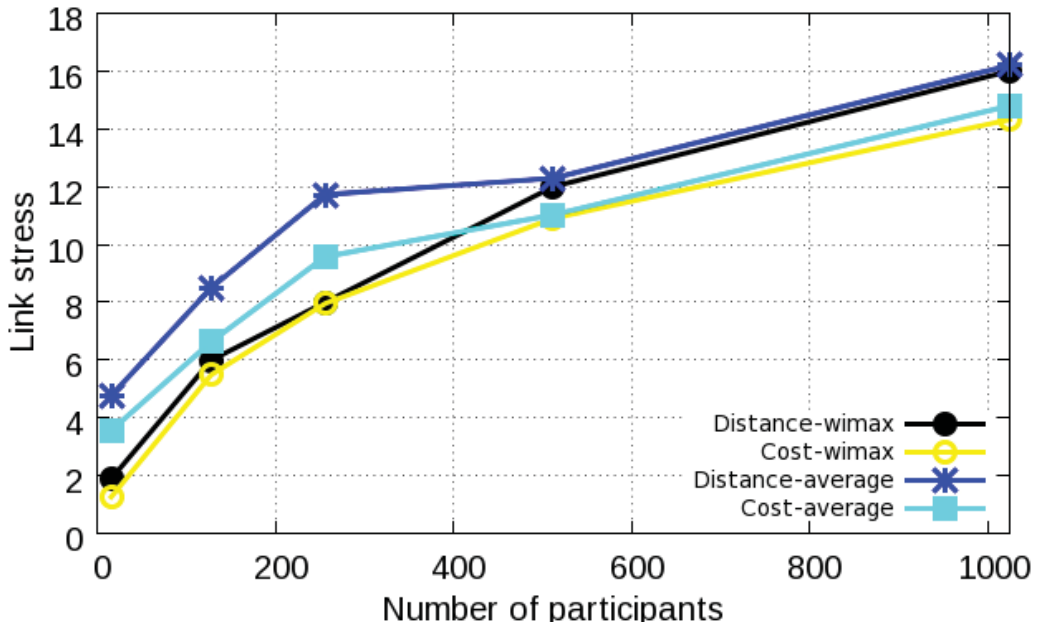


Figure 3.12: Link stress performance of the video conferencing service built over a convergent underlay network comprised of a simulated Internet topology and a real mobile WiMAX network.

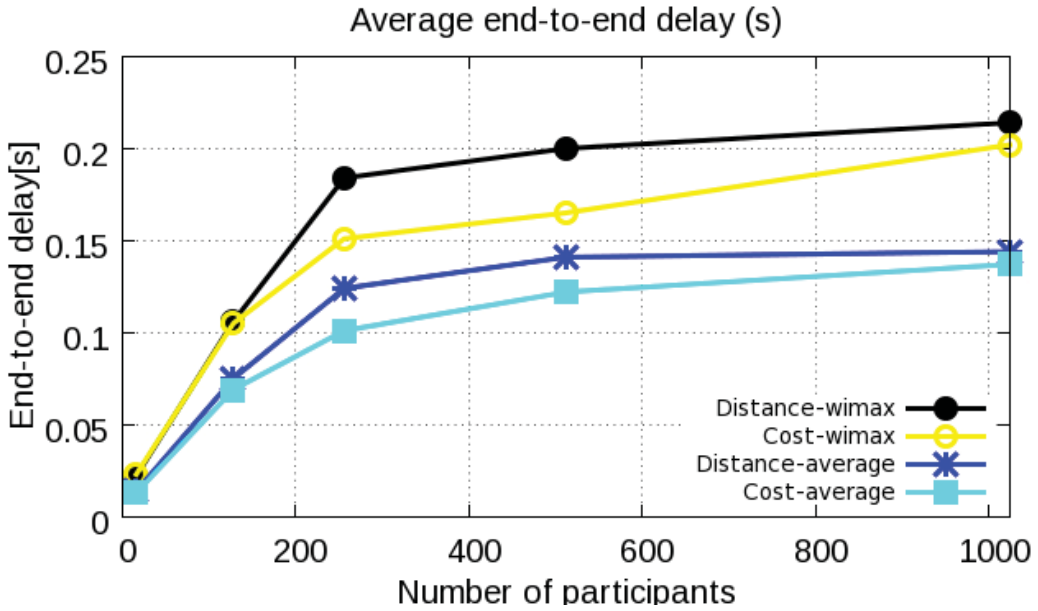


Figure 3.13: End to end delay performance of the video conferencing service built over a convergent underlay network comprised of a simulated Internet topology and a real mobile WiMAX network.

function in both the average link stress and the average end-to-end delay.
In comparison between the mobile WiMAX link and the entire overlay network built over the convergent underlay convergent network, Figure 3.12 shows that the mobile WiMAX link has a smaller link stress value than the average link stress on the entire overlay network. The reason is because the wireless link has less available bandwidth and a higher delay than other link within the overlay network. Therefore, the cost (calculated by our newly proposed cost function) for a data packet to travel through the mobile WiMAX links is high. The overlay routing algorithm will try to avoid as much as possible the real mobile WiMAX links and that is why they are placed at lower layers of the multicast tree. At the lower layers, the mobile WiMAX links and nodes do not have to do too much data forwarding for the ALM. Thus, they will have less link stress than other links in the overlay network. Our newly proposed cost function manage this automatically. Figure 3.13 shows that the end-to-end delay on the mobile WiMAX link is higher than the average end-to-end delay on the overlay network. It is normal since the wireless links generally have a higher delay than the fixed networks. Moreover, the wireless link here is a real WiMAX radio environment while the overlay network only takes simulation results.

3.6 Conclusion

In this chapter, a new cross-layer multi-variable cost function has been proposed. It gives a better optimum cost and provides a bigger set of suitable routes for the ALM algorithms to choose the optimal route from. The mathematical derivation process has also been described in details so that one can apply it to obtain other cross-layer multi-variable cost functions according to their specific requirements, in other contexts and in more general use cases. In this chapter, delay and bandwidth are used as an example but we also show how to form the cost function with more QoS parameters ($N \geq 3$) such as PER (Packet Error Rate), BER (Bit Error Rate), jitter...
Theoretical analysis has shown that the newly proposed cross-layer multi-variable cost function can simultaneously consider various requirements from the application as well as the possible resource from the network. It performs a cross layer joint optimization between the Application and Network layers. For the scalable video multicast and conferencing services running on the simulated overlay network, the new cost function outperforms the conventional distance function on both the average link stress and average end-to-end delay. Intensive simulation results built over the convergent underlay network of a simulated Internet topology and a real mobile WiMAX network have shown that, for a video conferencing service running on both the real mobile WiMAX link and the overlay network, the newly proposed cross-layer multi-variable cost function outperforms the conventional distance function in all four metrics of link stress, average link stress, end-to-end delay and average end-to-end delay.
If multicast is the one-to-many mechanism to distribute the multimedia content, a deeper study on the many-to-many mechanism will be done in the next part of the thesis through

a new architecture for video conferencing services.

Chapter 4

Enriched human perception-based distributed architecture for scalable video conferencing services: theoretical models and performance

Contents

4.1 Introduction **53**
4.2 Proposed Architecture of Human Perception-based Distributed Scalable Video Conferencing Services **54**
4.3 Theoretical analysis **56**
4.3.1 Theoretical expression of the waiting time in queues 61
4.3.2 Numerical calculation of the waiting time 68
4.3.3 Point-to-point delay 72
4.3.4 Required service rate 76
4.3.5 Estimation of global throughput in the network 78
4.4 Conclusion **80**

4.1 Introduction

This chapter proposes an enriched human perception-based distributed architecture for the multi-party video conferencing services. The newly proposed architecture can effectively reduce the unnecessary traffic of the multi-layer video streams transmitted on the overlay network. Rich theoretical models of the three different architectures: the proposed perception-based distributed architecture, the conventional centralized architecture and perception-based centralized architecture have been constructed by using queuing theory to reflect the traffic generated, transmitted and processed at the perception-based distributed leaders, the perception-based centralized top leader, and the centralized server. The performance of these three different architectures has been considered in different aspects from the total waiting time, the point-to-point delay and the required service rates to the total throughput. The theoretical models and analysis

give a better comparison results among the three different architectures. These results help the readers to have a global view of the proposal's performance in a fair comparison with the conventional methods. Together, the modelling tools, the analysis, and the numerical results help to answer the common concern about advantages and disadvantages between the centralized and distributed architectures for the video conferencing service architecture. Overall, the proposed human perception-based distributed architecture for multi-party video conferencing services can maintain a smaller total waiting time and point-to-point delay with a much smaller requirement of service rate and an equivalent total throughput in comparison with the conventional centralized architecture and perception-based centralized architecture.

4.2 Proposed Architecture of Human Perception-based Distributed Scalable Video Conferencing Services

In general, any cost function can be applied to build the media distribution tree for Scalable Video Coding contents. In our first proposal [47] and in this thesis, the application-aware multi-variable cost function proposed in [79] can be used as the optimal cost function to build the media distribution tree for the perception-based distributed architecture. The following model and analysis can be generally applied to a media distribution tree built from any cost function.

Figure 4.1 displays the main characteristics of our proposal. A cluster is a group of k peers which have the nearest distance to each other according to the optimization of the applied cost function. When a peer wants to join the overlay group, it will first try to explore its nearest cluster to join into by measuring the costs to reach the leaders of all clusters. A cluster has the maximum size of **k** peers depending on the network conditions. A group's leader is the one who has the minimum total cost to reach to all other peers in the cluster. Here we assume that all the important criteria such as processing and computational capacity, the available memory, bandwidth of the group's leader candidate have been fully considered in the applying multi-variable cost function before calculating the cost. All leaders, from the first layer, will then use the same cost function to calculate its costs to reach to all other leaders at layer 1. These costs will then be applied to form clusters and a second layer. The calculations are made until a the maximum number of layers is reached ($l_{max} = log_k N$). A leader l will receive bit-streams from its cluster's peers j with a throughput of $\lambda_{j\to l}$ and a traffic variation represented by $C^2_{j\to l}$. The leader makes (k-1) duplications of the arriving bit-stream before multicasting the traffic back to the other peers at a variation of $C^2_{\psi\to l}$ and to the upper layer's leaders (each peer within the cluster will receive bit-streams from all other peers in the same cluster but not its own bit-stream). At the same time, it receives the bit-streams from upper layer's leader and forwards them to its cluster's members.

The proposed architecture is applied when Scalable Video Coding (or other kinds of multilayer video coding) is used on the video conferencing service. The main advantage of multi-layer video coding in general and Scalable Video Coding in particular is that

the video can be encoded into a base layer bit-stream and several enhancement layer bit-streams.

We call the maximum pre-defined number of enhancement video layers at each participant n^e_{max}. Each participant can send an arbitrary number of enhancement video layers to the overlay group, but it should not excess the maximum number of enhancement video layers.

In a video conferencing session, at any given time, there are normally one or a few active speakers (active speakers are the conference participants who are giving the speech or actively participating into the argument/discussion). The active speaker can be automatically found by comparing the participant microphones' output power. A simple reason is that, if all participants are to be displayed with full quality in a conference session, a human-being will not have enough perception capacity to follow all of them. Their terminals also have the difficulty in displaying all of the full quality video bit-streams from too many users. From the multicast tree's point of view, an Auto Active Speaker Detector (AASD) can easily reduce the unnecessary traffic for the entire distributed system. The AASD is a functional block placed at each peer to automatically detect whether the peer is an active speaker or not by comparing its input audio powers [80], by visual information [81], or any combination of audio-visual methods [82][83].

We call $r^b_{hl}, r^{e_i}_{hl}$ the traffic rectification coefficients on base and i^{th} enhancement video layers from h^{th} participant to a cluster leader l. Let $r^{e_i}_{hl} = 1$ if the member wants to receive i^{th} enhancement video layer from h^{th} participant, it equals to 0 otherwise.

A peer sends its base layer SVC stream with a bit-rate of γ^b_{hl} to its cluster's leader. Beside sending base video layers, active speakers also send their i^{th} enhancement layers to the cluster's leader (at a bit-rate of γ^{ei}_{hl}). Enhancement layer bit-stream from an inactive but interesting users may be desirable for some peers. In this case, those particular peers can inform its cluster's leader about the interesting user(s) they want to receive enhancement video layers from. Through a network of leaders, this information will be notified to the interesting users and to all the cluster's leaders. After receiving the notification, the interesting users will send their enhancement video layers to the group as if they are active speakers. Each leader maintains a Conferee Preference Table (CPT) and to all the cluster's leaders. This is a record table of N rows and N columns (N is the number of participants). The AASD at each participant can also update its cluster leader whether it is an active speaker or not so that the cluster's leader can update to the Conferee Preference Table. Therefore, the default value of the Conferee Preference Table can be determined by the AASD at each participant. Another option for updating the values in the CPT is that each conferee can also select the interesting participant(s) it wants to receive enhancement video layers from by updating 1 to the corresponding place(s) (CPT[h,h']) of the table. Each participant can also select whether or not it wants to have the Conferee Preference Table automatically updated by its AASD. Thus, the Conferee Preference Table can be dynamically updated by the AASD or it can be manually maintained by the users' selections. The Conferee Preference Table's content is synchronized among all the leaders of the perception-based distributed architecture. As

a result, $r_{hl}^{ei} = 1$ if it is an active speaker detected by the AASD or it is an interesting user registered by at least one other participant, it equals to 0 otherwise. A participant sends its enhancement video layers if its corresponding traffic rectification coefficient $r_{hl}^{e_i} = 1$ and does not send it otherwise. On the other hand, after receiving the enhancement video layers from its upper layer leader, each leader decides whether or not it should forward an enhancement video layer to its cluster's members based on the updated information it has in its Conferee Preference Table. A private point to point video chat session can be established and maintained using the same Conferee Preference Table's mechanism.

Each peer in the multicast group will contribute only a portion of its computational capacity to support the conference according to the number of enhancement video layers required by the conference in order to maintain a steady-state of the multicast system (otherwise, there will be congestion at peers). This portion of contributing computational capacity is flexible and should be variable when more enhancement video layers are required. However, there are required service rates at the leaders (M_l) of the perception-based distributed architecture, at the central leader (M_p) of the perception-based centralized architecture and at the central MCU (M_m) of the centralized architecture. These are the required services rates to maintain steady-states at the queues processed by each architecture so that no congestion occurs.

These three different architectures will be mathematically modelled and analysed in the next section. The performance of the centralized, perception-based centralized and perception-based distributed architectures will be considered in terms of waiting time, end-to-end delay, required service rate, and total throughput. The waiting time performance compares the processing and queuing mechanisms of the three architectures. The end-to-end delay comparison considers the delay of a packet including the processing time in the queues and also the transmission time in the underlay network. The required service rates explain how much computational capacity the important nodes have to equip in order to support a steady state of the queue formed by each architecture. Finally, the total throughput performance answers a very common concern that whether the distributed architecture has to manage much more traffic than the centralized architecture or not. By approaching the performance from different aspects, a completed view of the proposal is archived.

4.3 Theoretical analysis

In order to compare the queuing delay of the three different centralized, perception-based centralized and perception-based distributed architectures, three queuing models are constructed with the notations in Table 4.1, Fig.4.2 and Fig.4.1.

Table 4.1: Scalable video codec parameters and notations for reference of the queuing model.

Video codec	H.264 Spatial SVC
Base video layer: QCIF	
Mean frame size ($\bar{X}_b$)	0.632 [kbyte]
Coefficient of variation of frame size (Cov_b)	1.757 [unit free]
Mean bit rate ($\bar{\gamma}_b$)	0.152 [Mbps]
SCV_b	16 [kbit]
Enhancement video layer: CIF	
Mean frame size ($\bar{X}_e$)	0.962 [kbyte]
Coefficient of variation of frame size (Cov_e)	2.043 [unit free]
Mean bit rate ($\bar{\gamma}_e$)	0.231 [Mbps]
Aggregated video: CIF	
Mean frame size ($\bar{X}_a$)	0.1.594 [kbyte]
Coefficient of variation of frame size (Cov_a)	1.9 [unit free]
Mean bit rate ($\bar{\gamma}_a$)	0.383 [Mbps]
SCV_a	46 [kbit]
N, k, n^e_{max}	Total number of participating peers, maximum cluster size, and the average of maximum number of enhancement video layers, respectively
$X^b_{hl}, X^{ei}_{hl}, X^p_{hl}$, X^p_{hp}, X^a_{hS}	Random inter-arrival time of the traffic generated by the base, i^{th} enhancement video layer, perception-based distributed aggregated, perception-based centralized aggregated, and centralized aggregated video layers from the participant h to the leader l of its cluster, to the centralized leader p, or to the centralized server S, respectively
$C^2_{Bl}, C^2_{Bp}, C^2_{Bs}$	Squared coefficient of variation (SCV) of service distributions at leader of layer l, at the top leader p of the perception-based centralized architecture, and MCU, respectively. These parameters depend on the hardware of processing nodes (peers, top leader, MCU)

Continued on next page

Table 4.1 – *Continued from previous page*

$C^2_{Aj}, C^2_{Al}, C^2_{Ap}, C^2_{AS}$	Squared coefficient of variation (SCV) of service distributions at each peer, leaders of layer l, the top leader p, and the MCU, respectively. These parameters depend on the variation of the *arriving* traffic to the peers, leaders, and MCU
$C^2_{Dj}, C^2_{Dl}, C^2_{Dp}, C^2_{Dm}$	Squared coefficient of variation (SCV) of service distributions at each peer, leaders of layer l, top leader p, and MCU, respectively. These parameters depend on the variation of the *departing* traffic from the peers, leaders, and MCU
$C^2_{ij}, C^2_{j\to S}, C^2_{j\to l}, C^2_{j\to p}, C^2_{\psi\to l}$	Squared coefficient of variation (SCV) of traffic from entity i to j, from peer j to the MCU, from peer j to a leader of layer l, from peer j to the top leader p, from the "multicast duplicator" to the leader of layer l, respectively
$C^2_{o\psi S}, C^2_{o\psi p}, C^2_{o\psi l}$	SCV of traffic from the "multicast duplicator" to the centralized MCU, the perception-based centralized leader, and the leader at layer l
γ^a_{hl}, γ^b_{hl}, γ^{ei}_{hl}, γ^p_{hl}, γ^p_{hp}, γ^a_{hS}, γ^a_h, γ^p_h	The aggregated, base, i^{th} enhancement video traffic generated by the SVC video encoder at h^{th} participant, the video traffic from the participant h to the leader l of its cluster, from the participant h to the perception-based centralized server, from the participant h to the centralized server (mcu), the video traffic from the participant h, the aggregated video traffic from the participant h, respectively
$\gamma_{\psi\to l}, \gamma_{\psi\to S}, \gamma_{\psi\to p}$	The external traffic from the "multicast duplicator" to the leader l, the centralized MCU, and the perception-based centralized leader, respectively
$\bar{\gamma}_b = E_h(\gamma^b_{hl})$, $\bar{\gamma}_e = E_h(\gamma^{ei}_{hl})$, $\bar{\gamma}_p = E_h(\gamma^p_{hp})$, $\bar{\gamma}_a = E_h(\gamma^a_{hS})$	Mean value for all of the conference participants of the base, enhancement, and aggregated video traffic from h^{th} participant to a leader in l^{th} layer and to the leader l of its cluster, respectively
$\lambda_{Al}, \lambda_{AS}, \lambda_{Ap}$	The throughput arriving to a leader at layer l, the centralized MCU, the perception-based centralized leader, respectively

Continued on next page

Table 4.1 – *Continued from previous page*

$\lambda_{i\to j}, \lambda_{j\to S}, \lambda_{j\to l}, \lambda_{j\to p}, \lambda_{j\to L}, \lambda_{L\to j}$	Throughput from entity i to j, from peer j to the MCU, from peer j to a leader of layer l, from j to the perception-based centralized leader, from member j to the centralized leader L and from the centralized leader L to the member j in the perception-based centralized architecture, respectively
$\lambda_j, \lambda_l, \lambda_m$	Throughput at the peer j, the leader l, and the MCU, respectively
$r_{hl}^b, r_{hl}^{ei}, r_{hp}^b, r_{hp}^{ei}$	Traffic rectification coefficients on base and i^{th} enhancement video layers from h^{th} participant to the leader l of its cluster, to the perception-based centralized server
$\bar{r}_b = E_h(r_{hl}^b)$, $\bar{r}_e = E_h(r_{hl}^{ei})$	Mean values of the traffic rectification coefficients for base, enhancement video traffics from h^{th} participant to a leader in l^{th} layer and to the leader l of its cluster, respectively
μ_l, μ_p, μ_m, M_l, M_p, M_m	Service rates and required service rates at the leaders on layer l of the perception-based distributed architecture, at the central leader of the perception-based centralized architecture and at the central MCU of the centralized architecture
ρ_l, ρ_p, ρ_S	Traffic intensity (traffic congestion) at leaders of layer l, at the top leader p, at the MCU, respectively ($\rho = \frac{\lambda}{\mu}$)

According to [84], the approximated waiting time of each service architecture (G/G/1 queue of General distribution of inter-arrival time, General distribution of service time, 1 parallel server) is calculated by:

$$W_q \approx \left(\frac{\rho}{1-\rho}\right)\left(\frac{C_A^2 + C_B^2}{2}\right)\left(\frac{1}{\mu}\right) g(\rho, C_A^2, C_B^2) \tag{4.1}$$

where:

$$g(\rho, C_A^2, C_B^2) = \begin{cases} exp\left[-\dfrac{2(1-\rho)}{3\rho}.\dfrac{(1-C_A^2)^2}{C_A^2 + C_B^2}\right], & \text{if } C_A^2 < 1 \\ 1, & \text{if } C_A^2 \geq 1 \end{cases} \tag{4.2}$$

Formula (4.1) has a nice "product form" of three terms: (i) a traffic-intensity factor; (ii) a variability factor; and (iii) a time-scale factor (each packet requires $\frac{1}{\mu}$[unit time] of service).

In formula (4.1), since the service distributions of the considered systems are not exponential (more precisely they follow general distributions), analytical results are not

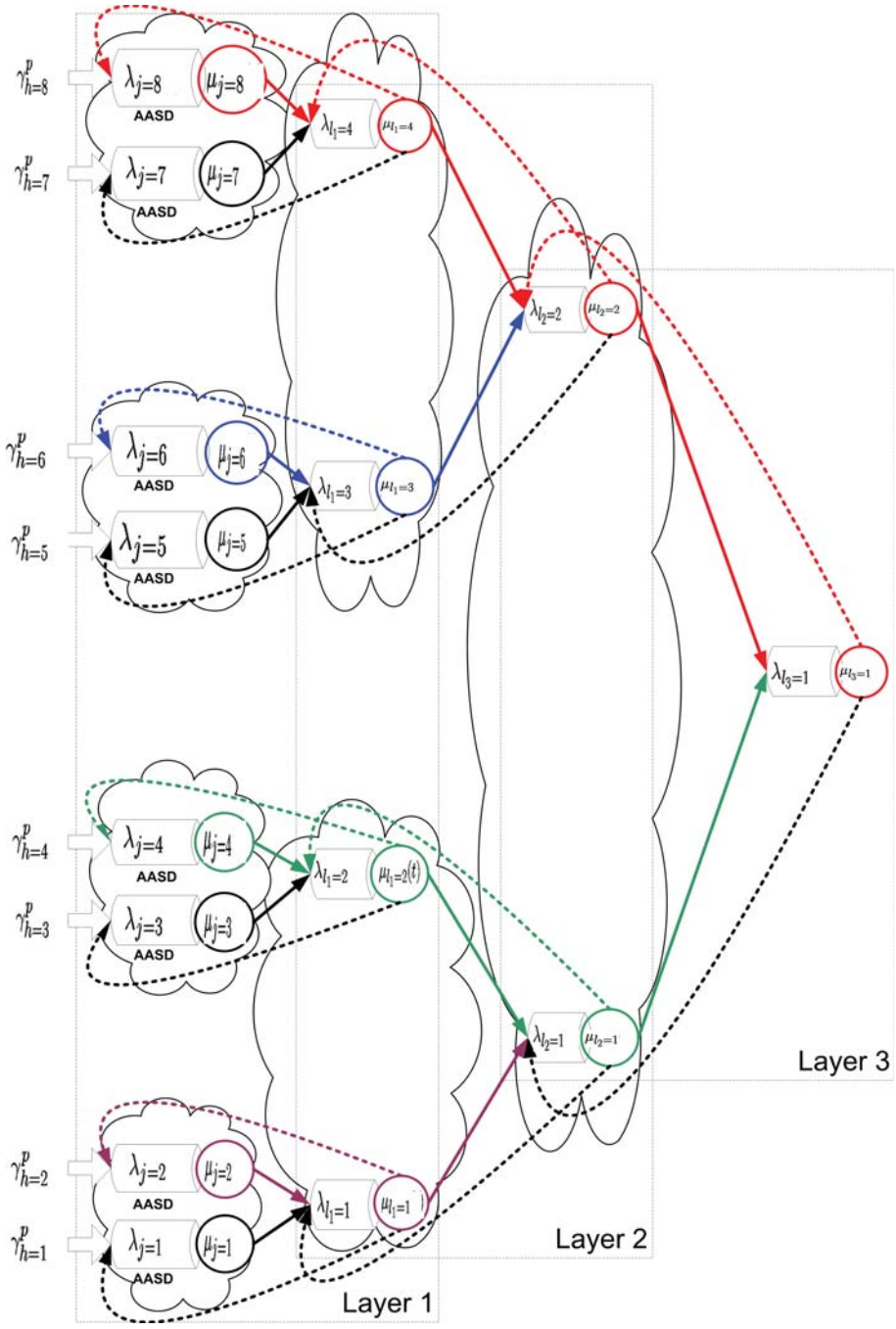


Figure 4.1: General model analysis of the perception-based distributed video conference service architecture when N=8, k=2.

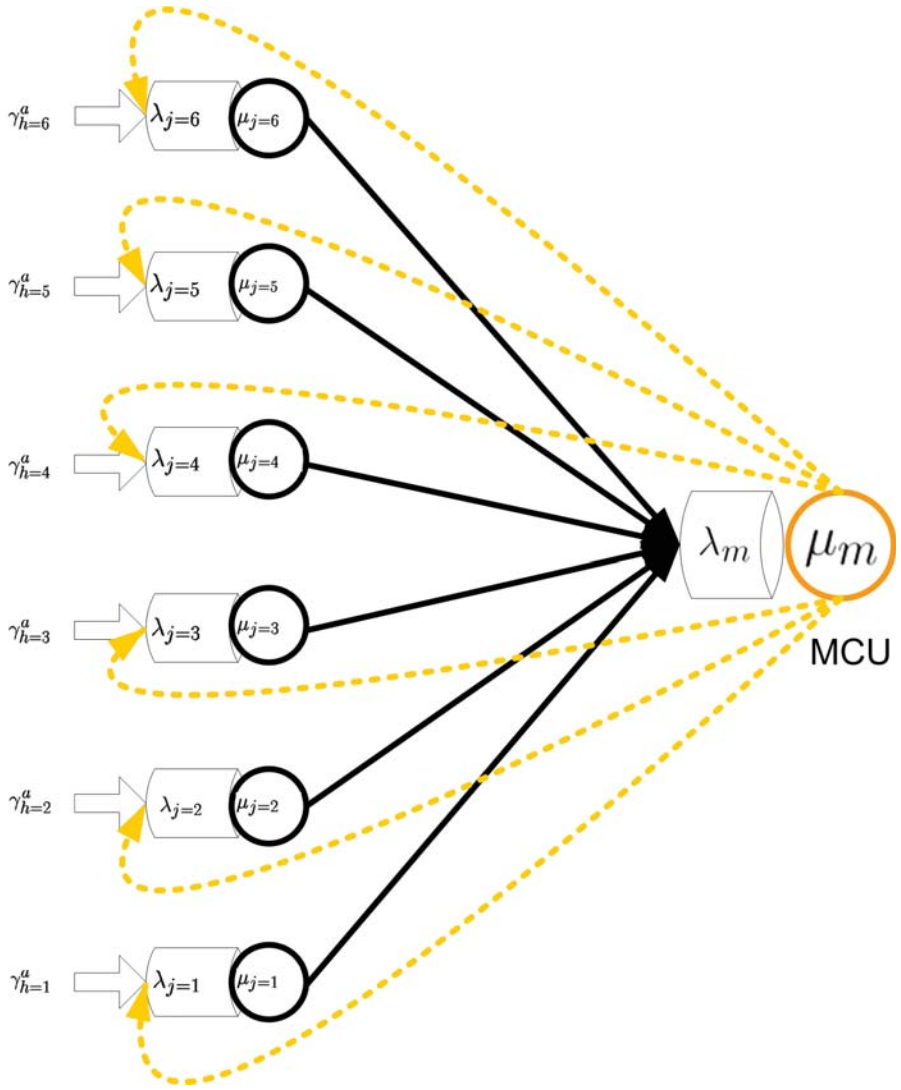


Figure 4.2: General model analysis of the centralized video conference service architecture when N=6.

generally achievable and so approximation methods are needed. There are several types of approximation, here we use a class of methods called decomposition methods. These methods consist of decomposing a network into smaller sub-networks and then analyzing these sub-networks as separate and independent entities. Typically, the sub-networks are individual queues.
Several assumptions are made for formula (4.1):

- The arrival process to each node can be approximated well by a renewal process.
- Service times at i are IID random variables with mean $\frac{1}{\mu}$, and squared coefficient of variation (SCV) C_{Bi}^2,
- All service times, external inter-arrival times, and routing transitions are independent of all else. They follow general distributions rather than being restricted to exponential.
- Each and every node is assumed to be operating below capacity.

C_B^2 is a fixed value determined by the type of hardware used at the queuing nodes. The SCV value of the distribution of arriving traffic at each node (C_{Ai}^2) can be calculated by:

$$C_{Ai}^2 = \frac{1}{\lambda_i}\left(\gamma_i C_{Oi}^2 + \sum_{j=1}^{k} \lambda_j C_{ji}^2\right) \tag{4.3}$$

The SCV value of the distribution of departing traffic from each node (C_{Di}^2) can be calculated by:

$$C_{Di}^2 = \rho_i^2 C_{Bi}^2 + (1-\rho_i^2) C_{Ai}^2 \tag{4.4}$$

Then,

$$C_{ij}^2 = \frac{\lambda_{i\to j}}{\lambda_{Di}} C_{Di}^2 + \left(1 - \frac{\lambda_{i\to j}}{\lambda_{Di}}\right) \tag{4.5}$$

Replace Equ.4.4 into Equ.4.5, we have the final form of the SCV of service rate for traffic generated by node i departing for node j:

$$C_{ij}^2 = \frac{\lambda_{i\to j}}{\lambda_{Di}} \left[\rho_i^2 C_{Bi}^2 + (1-\rho_i^2) C_{Ai}^2\right] + \left(1 - \frac{\lambda_{i\to j}}{\lambda_{Di}}\right) \tag{4.6}$$

For all of the following models, we use an abstract concept of traffic arriving from the "multicast duplicator". This is due to the fact that, in a multicast session, each transmitting node has to duplicate the arriving traffic before forwarding them to several multicast receivers. We model that extra traffic as the traffic from an abstract "multicast duplicator" (ψ) at the multicast transmitter.
Since the end-to-end delay of each architecture depends on the queuing time, if we succeed in building queuing models and then calculate the approximated waiting time for the three different architectures, end-to-end delay can be compared based on these results.

4.3.1 Theoretical expression of the waiting time in queues

Each peer encodes a scalable video stream with a base layer and several enhancement layers. To calculate the waiting time in the three architectures, we need to properly calculate the value of CoV^2 of the combined stream in these three cases from the available values of the base and enhancement video layers (component streams). We apply the derivation from [84]. This derivation is motivated by (a) approximating the component streams as independent renewal processes and (b) approximating the combined stream also as a renewal process.

Consider a node h in the network. Let $N_{jh}(t)$ be the number of arrivals from j to h that have occurred by time t (where $j = 0, 1, ..., k$ and N_{0h} represents the arrival process of the base video layer to h; N_{jh} with $j > 0$ represents the arrival process of the enhanced video layer to h. We assume that N_{jh} is a renewal process and that these renewal processes are independent of each other. Let X_{jh} be a random inter-arrival time of this process. Let $N_h(t) \equiv \sum_{j=0}^{n^e_{max}} N_{jh}(t)$ be the combined arrival process to h. We assume that $N_j(t)$ is also a renewal process. Let X_h denote a random inter-arrival time of this process. Define the following limits, which we assume to exist:

$$\begin{cases} \gamma_{jh} \equiv \lim\limits_{t\to\infty} \dfrac{E(N_{jh}(t))}{t} \\ \vartheta_{jh} \equiv \lim\limits_{t\to\infty} \dfrac{Var(N_{jh}(t))}{t} \end{cases} \tag{4.7}$$

That is, γ_{jh} is the average rate of packets departing from j arriving at h. It is also the average package rate from the j^{th} layer of the scalable video stream arriving at node h ($j = 0$ indicates the base video layer). Since we have assumed the component processes are independent (an approximation), we have:

$$\begin{cases} E(N_h(t)) = \sum\limits_{j=0}^{n^e_{max}} E(N_{jh}(t)) \\ Var(N_h(t)) = \sum\limits_{j=0}^{n^e_{max}} Var(N_{jh}(t)) \end{cases} \tag{4.8}$$

Then Equ.4.7 and Equ.4.8 imply that:

$$\begin{cases} \gamma_h = \sum\limits_{j=0}^{n^e_{max}} \gamma_{jh} \\ \vartheta_h = \sum\limits_{j=0}^{n^e_{max}} \vartheta_{jh} \end{cases} \tag{4.9}$$

where $\gamma_h \equiv \lim\limits_{t\to\infty} E(N_h(t))$ and $\vartheta_h \equiv \lim\limits_{t\to\infty} Var(N_h(t))$.

For a renewal process $N_h(t)$, there is a one-to-one relation between the first two moments

of the inter-arrival distribution, $E(X_h)$ and $E(X_h^2)$, and the limiting values of the renewal process, γ_h and ϑ_h. Specifically, this relationship has been proved in [85]:

$$\begin{cases} \gamma_h = \dfrac{1}{E(X_h)} \\ \vartheta_h = \dfrac{E(X_h^2) - E^2(X_h)}{E^3(X_h)} \end{cases} \tag{4.10}$$

From the relationship in Equ.4.10, we have the following:

$$C_{Ah}^2 \equiv \frac{Var(X_h)}{E^2(X_h)} = \frac{E(X_h^2) - E^2(X_h)}{E^2(X_h)} = \vartheta_h E(X_h) = \frac{\vartheta_h}{\gamma_h} \tag{4.11}$$

Similarly, the analogous relationship applies to each component renewal process namely, $C_{jh}^2 = \frac{\vartheta_{jh}}{\gamma_{jh}}$. Combining these results gives:

$$C_{Ah}^2 = \frac{\vartheta_h}{\gamma_h} = \frac{1}{\gamma_h} \sum_{j=0}^{n_{max}^e} \gamma_{jh}.C_{jh}^2 \tag{4.12}$$

where we have used that γ_{0h} is the arrival rate from the base video layer to h and γ_{jh} with $j > 0$ is the arrival rate from the enhancement video layer(s) to h.
According to [84], this method of approximating the arrival stream is based on an asymptotic method [86] in which the approximating renewal process is chosen so that the asymptotic values (e.g. $\lim_{t\to\infty} \frac{E(N_h(t))}{t}$ and $\lim_{t\to\infty} \frac{Var(N_h(t))}{t}$) of the approximating process match those of the original process. Many other approaches have also been suggested to approximate flows in a queue. For example, the stationary-interval method [86] chooses an approximating renewal process so that the moments from the renewal interval match the moments of the stationary distribution of an interval from the original process. Albin [87] and Whitt [88] give hybrid approaches the combine results from the asymptotic and stationary-interval methods.
From the derivation result presented in Equ.4.12, we can obtain the CoV value of the combined scalable video stream from the component streams of base and enhancement video layers.
In the case of perception-based distributed architecture, we have:

$$CoV^2(X_{hl}^p) = \frac{1}{\gamma_{hl}^a} \sum_{i=0}^{n_{max}^e} \gamma_{hl}^i.CoV^2(X_{hl}^{ei}) \tag{4.13}$$

In the case of perception-based centralized architecture, we have:

$$CoV^2(X_{hp}^p) = \frac{1}{\gamma_{hp}^a} \sum_{i=0}^{n_{max}^e} \gamma_{hp}^i.CoV^2(X_{hp}^{ei}) \tag{4.14}$$

In the case of centralized architecture, we have:

$$CoV^2(X_{hS}^a) = \frac{1}{\gamma_{hS}^a} \sum_{i=0}^{n_{max}^e} \gamma_{hS}^i.CoV^2(X_{hS}^{ei}) \tag{4.15}$$

4.3.1.1 Perception-based distributed architecture

For the perception-based distributed architecture, the distributed model is as shown in Fig.4.1. The throughput arriving to a leader at layer l (λ_{Al}) is comprised of:

- Throughput arriving from all peers j of the cluster to the leader l ($\lambda_{j\to l}$),
- Throughput from the "multicast duplicator" ψ_l to the leader l ($\gamma_{\psi\to l}$). It is considered as the external traffic at the leader l.

The throughput arriving from all peer j of the cluster to the leader l ($\lambda_{j\to l}$) is comprised of the throughput from all the video bit-streams from all the conferees belonging to the sub-trees whose root is the leader l to the leader l:

$$\lambda_{j\to l} = \sum_{h=1+(j-1)k^{l-1}}^{jk^{l-1}} \gamma_{hl}^{p} \tag{4.16}$$

The throughput from the "multicast duplicator" ψ_l to the leader l ($\gamma_{\psi\to l}$) comprises of (k-1) copies of throughput from all the peers:

$$\gamma_{\psi\to l} = (k-1)\sum_{h=1}^{N} \gamma_{hl}^{p} \tag{4.17}$$

From Equations (4.16) and (4.17), we can calculate the overall throughput arriving to the leader l:

$$\lambda_{Al} = k\sum_{h=1}^{N} \gamma_{hl}^{p} \tag{4.18}$$

The purpose is to form an equation for calculating the SCV of times between arrivals to the leader l because it will determine the waiting time of the traffic queue at the leader l. The general form of C_{Al}^2 is:

$$C_{Al}^2 = \frac{v_{Al}}{\lambda_{Al}} \tag{4.19}$$

In which, v_{Al} is calculated by:

$$\vartheta_{Al} = \gamma_{\psi\to l}.C_{o\psi l}^2 + \sum_{j=1}^{N} \lambda_{j\to l}.C_{j\to l}^2 \tag{4.20}$$

Applying Equ.4.3, Equ.4.19 and Equ.4.20, we can obtain the SCV of the service rate at the leader l:

$$C_{Al}^2 = \frac{1}{\lambda_{Al}}.\left(\gamma_{\psi\to l}.C_{o\psi l}^2 + \sum_{j=1}^{N} \lambda_{j\to l}.C_{j\to l}^2\right) \tag{4.21}$$

We need to calculate the SCV of times between departures from peer j to the leader l ($C_{j\to l}^2$) and from the abstract "multicast duplicator" to the leader l ($C_{o\psi l}^2$).

The general form of the SCV of times between departures from the abstract "multicast duplicator" to the leader l ($C^2_{o\psi l}$) is:

$$C^2_{o\psi l} = \sum_{j}^{k-1} C^2_{j\to l} \tag{4.22}$$

At the first layer, since all the peers are directly the video conferees, we have:

$$C^2_{j\to l=1} = C^2_{h=j\to l} = CoV^2(X^p_{hl}) \tag{4.23}$$

The value of $CoV^2(X^p_{hl})$ is given by Equ.4.13.
We can calculate the SCV of times between departures from the abstract "multicast duplicator" to the leader l as the combination of the SCV of times between departures from (k-1) peers j to the leader l:

$$C^2_{o\psi l=1} = \sum_{j=1}^{k-1} C^2_{j\to l=1} \tag{4.24}$$

In Equ.4.21, for the first layer, all of the parameters have been given in Equ.4.18 (λ_{Al}), Equ.4.17 ($\gamma_{\psi\to l}$), Equ.4.16 ($\lambda_{j\to l}$), Equ.4.24 ($C^2_{o\psi l}$), and Equ.4.23 ($C^2_{j\to l}$). Therefore, $C^2_{Al=1}$ is known for the first layer.
At the second layer, according to Equ.4.4 and Equ.4.5, we have:

$$C^2_{j\to l=2} = \frac{\lambda_{j\to l}}{\lambda_{Dj}}\left[\rho_j^2.C^2_{Bj} + (1-\rho_j^2).C^2_{Aj}\right] + (1 - \frac{\lambda_{j\to l}}{\lambda_{Dj}}) \tag{4.25}$$

The traffic intensity (traffic congestion) at each peer (ρ_j) is calculated by:

$$\begin{cases} \rho_j = \dfrac{\lambda_j}{M_l} \\ \lambda_j = \lambda_{Aj} = k\sum\limits_{h=1}^{N}\gamma^p_{hl} \end{cases} \tag{4.26}$$

λ_{Dj} is the throughput departing from a peer j, calculated by:

$$\lambda_{Dj} = k.\sum_{h=1}^{N}\gamma^p_{hl} \tag{4.27}$$

For the second layer, the SCV of service rate for the traffic arriving to the peer j (C^2_{Aj}) at layer 2 can be calculated from $C^2_{Al=1}$ at the first layer as:

$$C^2_{Aj} = C^2_{Al=1} \tag{4.28}$$

In Equ.4.25, all of the parameters are known ($\lambda_{j\to l}$ in Equ.4.16, λ_{Dj} in Equ.4.27, ρ_j in Equ.4.26, C^2_{Aj} in Equ.4.28). Therefore, $C^2_{j\to l=2}$ is known.

We can calculate the SCV of times between departures from the abstract "multicast duplicator" to the leader l of the second layer as the combination of the SCV of times between departures from (k-1) peers j to the leader l at the second layer:

$$C^2_{o\psi l=2} = \sum_{j=1}^{k-1} C^2_{j\to l=2} \tag{4.29}$$

In Equ.4.21, for the second layer, all of the parameters have been given in Equ.4.18 (λ_{Al}), Equ.4.17 ($\gamma_{\psi\to l}$), Equ.4.16 ($\lambda_{j\to l}$), Equ.4.29 ($C^2_{o\psi l=2}$), and Equ.4.25 ($C^2_{j\to l=2}$). Then, we would be able to calculate the SCV of service rate for the traffic arriving to the leader l at layer 2 ($C^2_{Al=2}$).
Recursively, we can calculate the values of C^2_{Al} on all the upper layers by applying the same method of calculating it on the first and the second layers and using Equ.4.21.
From the value of C^2_{Al} of all layers obtained from Equ.4.21 we can calculate the waiting time at a leader in layer l according to Equ.4.1, we have:

$$W_{ql} \approx \left(\frac{\rho_l}{1-\rho_l}\right)\left(\frac{C^2_{Al}+C^2_{Bl}}{2}\right)\left(\frac{1}{\mu_l}\right) \tag{4.30}$$

The required service rate and the congestion rate at each leader l (μ_l, ρ_l) are explained and calculated in more details in section 4.3.4 and Equ.4.60.
Since k members in a cluster have to wait in sequence to be served by the cluster's leader, the total waiting time of the perception-based distributed architecture is calculated by:

$$W_{qd} = \sum_{l=1}^{l_{max}} kW_{ql} \tag{4.31}$$

In which l_{max} is the maximum number of layers in the perception-based distributed architecture.

4.3.1.2 Centralized architecture

For the centralized architecture, the centralized model is as shown in Fig.4.2. The throughput arriving to the centralized MCU (λ_{AS}) is comprised of:

- Throughput arriving from each peer $j = h$ to the MCU ($\lambda_{j\to S}$),
- Throughput generated by the "multicast duplicator" ψ_S for the MCU ($\gamma_{\psi\to S}$). It is considered as the external traffic at the MCU.

The overall throughput arriving to the centralized MCU is therefore:

$$\lambda_{AS} = \gamma_{\psi\to S} + \sum_{j=1}^{N} \lambda_{j\to S} \tag{4.32}$$

Applying Equ.4.3, we can obtain the SCV of the service rate at the MCU server:

$$C_{AS}^2 = \frac{1}{\lambda_{AS}}.\left(\gamma_{\psi\to S}.C_{o\psi S}^2 + \sum_{j=1}^{N}\lambda_{j\to S}.C_{j\to S}^2\right) \tag{4.33}$$

In which:

$$\begin{cases} C_{o\psi S}^2 = \sum_{j=1}^{N-1} C_{j\to S}^2 \\ C_{j\to S}^2 = CoV^2(X_{h=j,l=S}^a) = CoV^2(X_{hS}^a) \\ \gamma_{\psi\to S} = (N-2)\sum_{h=1}^{N}\gamma_{hS}^a \\ \lambda_{j\to S} = \gamma_{h=j}^a \\ \lambda_{AS} = (N-1)\sum_{h=1}^{N}\gamma_{hS}^a \end{cases} \tag{4.34}$$

From the value of C_{AS}^2 obtained from Equ.4.33 and Equ.4.34 we can calculate the waiting time at the MCU according to Equ.4.1, we have:

$$W_{qm} \approx \left(\frac{\rho_S}{1-\rho_S}\right)\left(\frac{C_{AS}^2 + C_{BS}^2}{2}\right)\left(\frac{1}{\mu_m}\right) \tag{4.35}$$

The required service rate and the congestion rate at the MCU (μ_m, ρ_m) are explained and calculated in more details in section 4.3.4 and Equ.4.61.
Since N participants have to wait in sequence to be served by the MCU, the total waiting time of the centralized architecture is calculated by:

$$W_{qc} = NW_{qm} \tag{4.36}$$

4.3.1.3 Perception-based centralized architecture

The perception-based centralized architecture can be considered as a special case of the perception-based distributed architecture in which the cluster size is equal to the total number of participants. Therefore, there is only one single cluster of (N-1) cluster members and one cluster leader. The perception-based centralized architecture can also be seen as a special case of the centralized architecture in which the participant can have an option to receive both the base and enhancement video layers (full quality) if the content is interesting and only the base video layer if the content is not.
The throughput arriving to the perception-based centralized server p (λ_{Ap}) is comprised of:

- Throughput arriving from each peer j to the perception-based centralized server p ($\lambda_{j\to p}$),
- Throughput generated by the "multicast duplicator" ψ_p to the top leader ($\gamma_{\psi\to p}$). It is considered as the external traffic at the perception-based centralized server p.

The overall throughput arriving to the perception-based centralized server p is therefore:

$$\lambda_{Ap} = \gamma_{\psi \to p} + \sum_{j=1}^{N} \lambda_{j \to p} \tag{4.37}$$

Applying Equ.4.3, we can obtain the SCV of the service rate at the perception-based centralized server p:

$$C_{Ap}^2 = \frac{1}{\lambda_{Ap}} . \left(\gamma_{\psi \to p} . C_{o\psi p}^2 + \sum_{j=1}^{N} \lambda_{j \to p} . C_{j \to p}^2 \right) \tag{4.38}$$

In which

$$\begin{cases} C_{o\psi p}^2 = \sum_{j=1}^{N-1} C_{j \to p}^2 \\ C_{j \to p}^2 = CoV^2(X_{h=j,l=p}^p) = CoV^2(X_{hp}^p) \\ \gamma_{\psi \to p} = (N-2) \sum_{h=1}^{N} \gamma_{hp}^p \\ \lambda_{j \to p} = \gamma_{h=j}^p \\ \lambda_{Ap} = (N-1) \sum_{h=1}^{N} \gamma_{hp}^p \end{cases} \tag{4.39}$$

After obtaining C_{Ap}^2 from Equ.4.38 and Equ.4.39, we can calculate the waiting time at the perception-based centralized server p according to Equ.4.1, since N members in a cluster have to wait in sequence to be served by the cluster's leader, the total waiting time of the perception-based centralized architecture is calculated by:

$$W_{qp} \approx N \left(\frac{\rho_p}{1 - \rho_p} \right) \left(\frac{C_{Ap}^2 + C_{Bp}^2}{2} \right) \left(\frac{1}{\mu_p} \right) \tag{4.40}$$

The required service rate and congestion rate at the perception-based centralized server p (μ_p, ρ_p) are explained and calculated in more details in section 4.3.4 and Equ.4.62.

4.3.2 Numerical calculation of the waiting time

The newly proposed theoretical models and mathematical expressions formed in the previous subsection 4.3.1 allow us to evaluate the performance in the four different performance criteria for the three different architectures in real-time, with an arbitrary number of participants, and with very heterogeneous contexts of peers such as peers with terminals of different screen sizes, different available bandwidth, different computational capacities and a variety of users' preferences. The proposed theoretical models and mathematical expressions can be applied to model a very complex enriched video conferencing services. The performance of these enriched video conferencing services can

be evaluated in real-time and for three different architectures over four different performance criteria. Our first step here is to have a first comparison of the three different architectures on the four different criteria. Therefore, we apply the available traffic models of the SVC video streams which are currently limited to the mean values of a video session presented in [89, 90] for a simple case of the distributed scalable video conference service. In this subsection, to provide first numerical results of the waiting time in the three different architectures, we apply an averaging of some video conferencing sessions' values presented in the previous subsection 4.3.1 so that we can apply the data provided in [89, 90]. The specifications of the spatial video traffic are shown in Table 4.1.

4.3.2.1 Perception-based distributed architecture

Figure 4.1 shows the queuing model of a random cluster on layer l in the perception-based distributed architecture. Each cluster has k peers, one of them is elected to be the cluster's leader based on its cost to reach all cluster's members. We apply the following assumptions for calculating the total waiting time of the perception-based distributed architecture:

- $\lambda_{j\to l} = k^{l-1}\bar{\gamma}_p$,
- $\gamma_{\psi\to l} = (k-1)N\bar{\gamma}_p$,
- $\lambda_{Al} = kN\bar{\gamma}_p$,
- $\rho_S = \dfrac{\lambda_{Al}}{\mu_l}$,
- $C^2_{o\psi\to l} = (k-1)C^2_{j\to l} = (k-1)CoV^2(X^p_{h=j})$,
- $C^2_{j\to l} = CoV^2(X^p_{h=j}) = CoV^2_p$ at first layer and $C^2_{j\to l} = C^2_{Al-1}$ at higher layers,
- $C_{Bl} = 0$ to avoid the effect of hardware on the queuing time,

Replace all the above calculations into Equ.4.21, we have:

$$\begin{cases} C^2_{Al} = \dfrac{CoV^2_p\left[(k-1)^2 + k^{l-1}\right]}{k} & \text{at first layer} \\ C^2_{Al} = \dfrac{(k-1)^2 CoV^2_p + k^{l-1}C^2_{Al-1}}{k} & \text{at higher layers} \end{cases} \tag{4.41}$$

From the value of C^2_{Al} obtained from Equ.4.41 we can calculate the waiting time at a leader in layer l according to Equ.4.1, we have:

$$W_{ql} \approx \left(\frac{\rho_l}{1-\rho_l}\right)\left(\frac{C^2_{Al} + C^2_{Bl}}{2}\right)\left(\frac{1}{\mu_l}\right) \tag{4.42}$$

The required service rate and the congestion rate at each leader l (μ_l, ρ_l) are explained and calculated in more details in section 4.3.4 and Equ.4.60.

Since k members in a cluster have to wait in sequence to be served by the cluster's leader, the total waiting time of the perception-based distributed architecture is calculated by:

$$W_{qd} = \sum_{l=1}^{l_{max}} kW_{ql} \tag{4.43}$$

In which l_{max} is the maximum number of layers in the perception-based distributed architecture.

4.3.2.2 Centralized architecture

Figure 4.2 shows the queuing model for the MCU-based architecture. Here, all N peers are generating a media stream with a mean data rate of $\bar{\gamma}_a$[Mbps] (this is the mean over the entire video conference session of the video traffic generated by the video participant j, $\bar{\gamma}_a = E_h(\gamma^a_{hl})$). Each peer sends all its encoded video to the MCU (e.g. peer's output is all sent to the common MCU). The MCU then routes back N media streams to N participating peers, each contains data from (N-1) other peers (assuming that each peer will not receive its own stream).
We apply the following assumptions for calculating the total waiting time of the centralized architecture:

- The external traffic found at the MCU is the duplicated traffic it generates for multicast purpose. This is modelled by the throughput from the multicast duplicator to the MCU ($\gamma_{\psi \to S}$) and its service variation ($C^2_{o\psi S}$),
- Mean value of the aggregated video traffic generated over time by the video encoder at participant h is $E_h(\bar{\gamma}^a_{hl}) = \bar{\gamma}_a$. The value of $\bar{\gamma}_a$ can be found in Table 4.1,
- $\lambda_{j \to S} = \bar{\gamma}_a$,
- $\gamma_{\psi \to S} = N(N-2)\bar{\gamma}_a$,
- $\lambda_{AS} = N(N-1)\bar{\gamma}_a$,
- $\rho_S = \dfrac{\lambda_{AS}}{\mu_m}$,
- $C^2_{o\psi S} = (N-1)C^2_{j \to S} = (N-1)CoV^2(X^a_{h=j})$,
- $C^2_{j \to S} = CoV^2(X^a_{h=j}) = CoV^2_a$. This can be obtained from Table 4.1,
- $C_{BS} = 0$ to avoid the effect of hardware on the queuing time,

Replace all the above calculations into Equ.4.33, we have:

$$C^2_{AS} = \frac{CoV^2_a\left[(N-1)(N-2)+1\right]}{(N-1)} \tag{4.44}$$

From the value of C_{AS}^2 obtained from Equ.4.44 we can calculate the waiting time at the MCU according to Equ.4.1, we have:

$$W_{qm} \approx \left(\frac{\rho_S}{1-\rho_S}\right)\left(\frac{C_{AS}^2 + C_{BS}^2}{2}\right)\left(\frac{1}{\mu_m}\right) \tag{4.45}$$

The required service rate and the congestion rate at the MCU (μ_m, ρ_m) are explained and calculated in more details in section 4.3.4 and Equ.4.61.
Since N participants have to wait in sequence to be served by the MCU, the total waiting time of the centralized architecture is calculated by:

$$W_{qc} = NW_{qm} \tag{4.46}$$

4.3.2.3 Perception-based centralized architecture

When the cluster size of the perception-based distributed architecture equals to the total number of participants ($k = N$), we obtain a new perception-based centralized architecture. The advantage of this new architecture is that it can inherit the perception-based ideas of the proposed perception-based distributed architecture such as the flexibility of reducing the unnecessary traffic due to the human-perception limitation. It requires a good leader node to manage all participants. We can obtain the parameters of the perception-based centralized architecture by replacing $k = N$ into the parameter set of the perception-based distributed architecture in Sec.4.3.2.1.

- $\lambda_{j\to p} = \bar{\gamma}_p$,
- $\gamma_{\psi\to p} = N(N-2)\bar{\gamma}_p$,
- $\lambda_{Ap} = N(N-1)\bar{\gamma}_p$,
- $\rho_p = \dfrac{\lambda_{Ap}}{\mu_p}$,
- $C_{o\psi p}^2 = (N-1)C_{j\to p}^2 = (N-1)CoV^2(X_{h=j}^p)$,
- $C_{j\to p}^2 = CoV^2(X_{h=j}^p) = CoV_p^2$,
- $C_{Bp} = 0$ to avoid the effect of hardware on the queuing time,

The SCV of service rate for the perception-based centralized architecture is therefore:

$$C_{Ap}^2 = \frac{CoV_p^2\left[(N-1)(N-2)+1\right]}{(N-1)} \tag{4.47}$$

After obtaining C_{Ap}^2 from Equ.4.47 we can calculate the waiting time at the perception-based centralized server p according to Equ.4.1, since N members in a cluster have

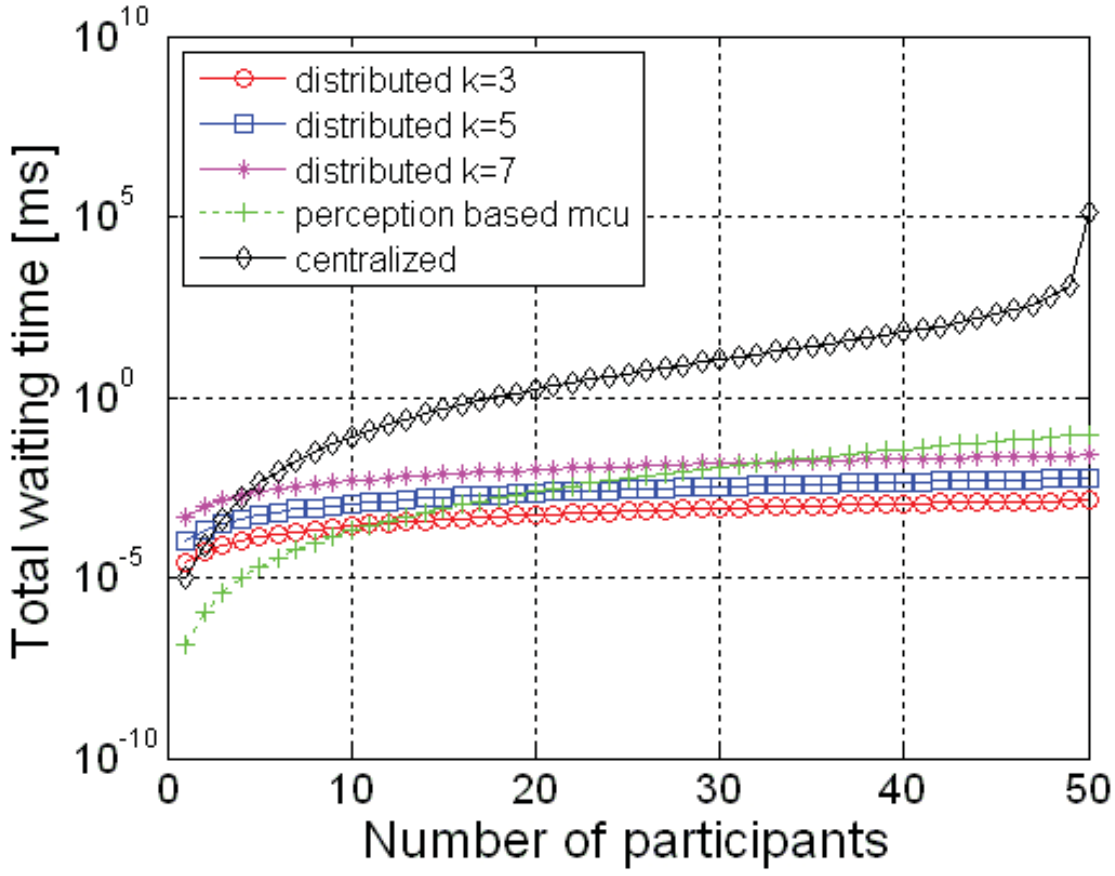


Figure 4.3: Comparison of queuing waiting time among centralized, perception-based centralized and perception-based distributed architectures at minimum traffic when $r_b = 1, r_e = 0.1, n_e = 3$.

to wait in sequence to be served by the cluster's leader, the total waiting time of the perception-based centralized architecture is calculated by, we have:

$$W_{qp} \approx N \left(\frac{\rho_p}{1 - \rho_p} \right) \left(\frac{C_{Ap}^2 + C_{Bp}^2}{2} \right) \left(\frac{1}{\mu_p} \right) \tag{4.48}$$

The required service rate and congestion rate at the perception-based centralized server p (μ_p, ρ_p) are explained and calculated in more details in section 4.3.4 and Equ.4.62.

4.3.2.4 Results Analysis

Figures 4.3, 4.4, 4.5 show the comparison among the total waiting time of the centralized architecture, the perception-based centralized architecture and the perception-based

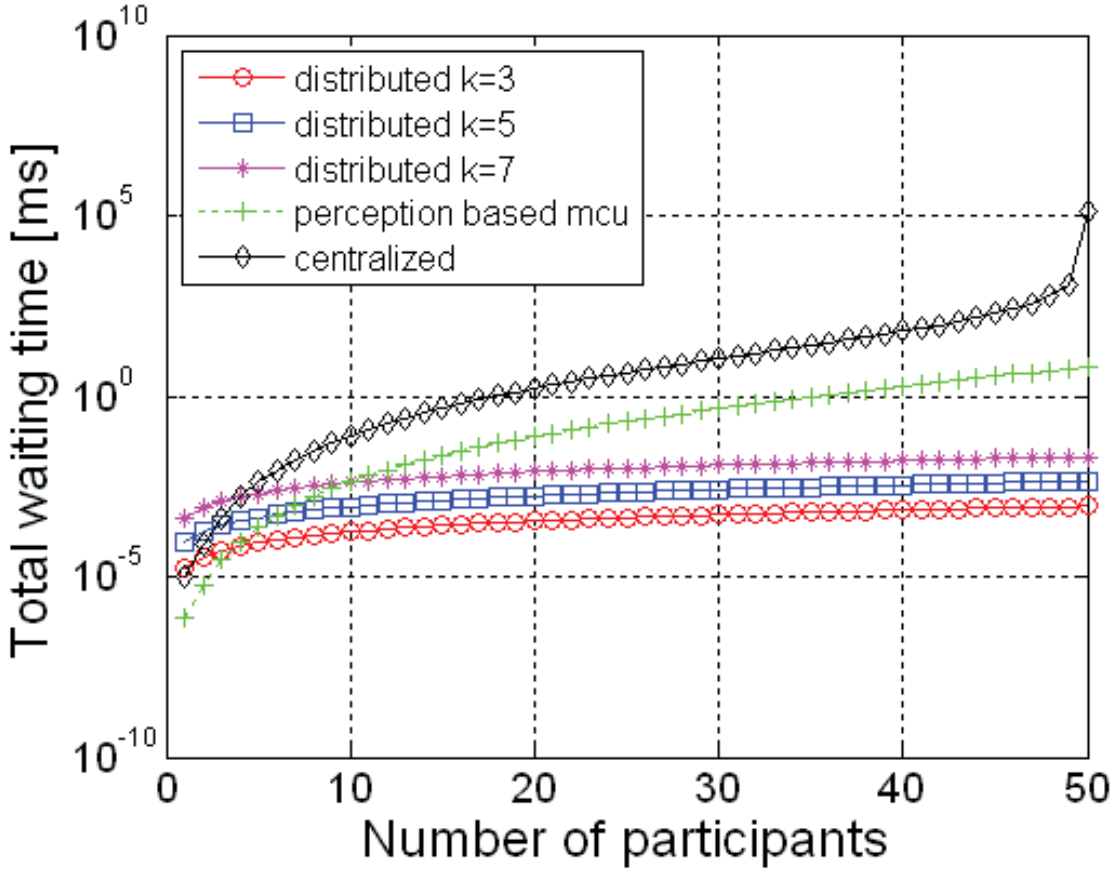


Figure 4.4: Comparison of queuing waiting time among centralized, perception-based centralized and perception-based distributed architectures at reduced traffic when $r_b = 1, r_e = 0.5, n_e = 3$.

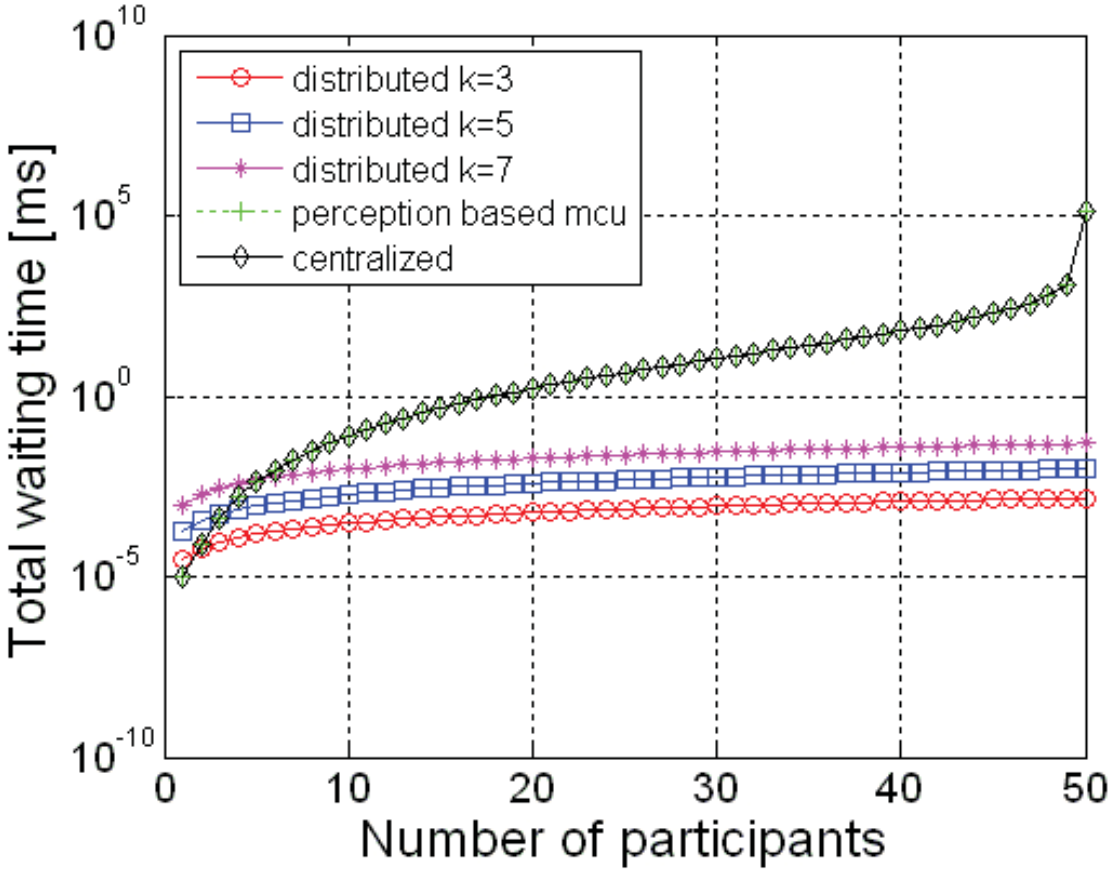


Figure 4.5: Comparison of queuing waiting time among centralized, perception-based centralized and perception-based distributed architectures at full traffic when $r_b = 1, r_e = 1, n_e = 3$.

distributed architecture when the clusters' sizes are $k = 3$, $k = 5$, $k = 7$. The video streams are encoded with spatial SVC and three different configurations of enhancement video traffic are considered. In the centralized architecture, we assume that the MCU can support of up to N_{max} participants at the same time, all participants are sending both its base and enhancement video layers. In the three figures, there are in average 3 enhancement video layers which are transmitted from all the participants ($n_e = 3$). In the perception-based centralized architecture and the perception-based distributed architecture, all peers send their base layers and some peers send their enhancement video layers. Each leader can support at least k peers in its cluster. We are going to analyze the results to show the effect of three main aspects on the total waiting time performance: (i) comparison between the distributed and the centralized architectures, (ii) impacts of the cluster size on the performance, and (iii) the newly proposed perception-based function's performance.

Regarding the waiting time performance comparison between the centralized and distributed architectures, we can compare the waiting time performance between the centralized and perception-based centralized architectures with the perception-based distributed architecture. The total waiting time of both the centralized and perception-based centralized architectures increases exponentially with the increasing number of participants. Meanwhile the total waiting time of the perception-based distributed architecture increases at a much lower logarithmic speed. In Fig.4.5, when full enhancement video traffic is sent to the perception-based centralized server, the perception-based centralized architecture is identical to the centralized architecture. In this figure, we can clearly see the gain in the total waiting time performance between the centralized and distributed architectures. The centralized architecture has the highest total waiting time followed by the perception-based centralized architecture. The perception-based distributed architecture outperforms the other two in terms of the total waiting time.

The cluster size also plays a role in the waiting time. There is an interesting conclusion which we can withdraw after analyzing the results in Figures 4.3, 4.4 and 4.5: when the number of peers in a cluster (k) increases, the total waiting time in the distributed queue increases. Therefore, for a certain number of participants (N=50), it is recommended to use a smaller cluster size to maintain a lower total waiting time.

When we make comparisons among the total waiting time of the three different figures ($r_e = 0.1$, $r_e = 0.5$, and $r_e = 1$), the more unnecessary traffic is reduced by applying our newly proposed perception-based function (both in the perception-based centralized and perception-based distributed architectures), the lower the total waiting time is. The newly proposed perception-based function can actually limit the unnecessary traffic and reduce the total waiting time.

From the results, we can conclude that the distributed architecture gives a smaller queuing waiting time than the two centralized architectures. Especially, when the number of participants increases, the distributed architecture outperforms the two centralized architectures in terms of the queuing waiting time. The proposed mathematical models and expressions can be used for determining the optimal value of the cluster size to

minimize the total waiting time. As the waiting time depends on the total traffic transmitted on the overlay network, by reducing the unnecessary traffic using our proposed human perception-based function, we can greatly reduce the total waiting time of the multi-party video conferencing service.

4.3.3 Point-to-point delay

We apply the point-to-point delay analysis proposed in [91] to analyse the delay caused by packet transmission on the underlay network. This large scale delay measurement study uses packet traces captured from an operational tier-1 network (point-to-point or router-to-router delay). The point-to-point delay comprises of:

- Propagation delay: depending on the physical characteristic of the media such as the medium and length,
- Transmission delay: depending on the link capacity, packet size,
- Nodal processing delay: depending on packet header examination, routing, check sum, transferring from input to output ports,
- Queuing delay: depending on the traffic load.

The above mentioned delays can be further grouped into fixed delay and variable delay:

$$d_{point-to-point} = d^{fix} + d^{var} \tag{4.49}$$

$$d^{fix}(p) = p.\sum_{i=1}^{h}(\frac{1}{C_i}) + \sum_{i=1}^{h}\delta_i \tag{4.50}$$

In which:

- h: Number of hops,
- C_i: Capacity of each link,
- δ_i: Propagation delay on each link,
- p: Packet's size.

Replacing $\alpha = \sum_{i=1}^{h}(\frac{1}{C_i})$ and $\beta = \sum_{i=1}^{h}\delta_i$ into Equ.4.50 we have:

$$d^{fix}(p) = p.\alpha + \beta \tag{4.51}$$

Through real measurement data:

- α is found to be in the range of $\{27.10^{-6} : 6.10^{-5}\}$,
- β is found to be in the range of $\{28 : 35\}$.

There are two value ranges for the d^{var}:

- If link utilization $\leq 90\%$ then $d^{var} \leq 1ms$,
- If link utilization $> 90\%$ then d^{var} is tens of milliseconds, we choose d^{var} to be $20ms$.

With a packet size ranging from $\{40 : 1500\}$ bytes (most common packet size on the Internet), we obtain d^{fix} from Equ.4.51.

4.3.3.1 Perception-based distributed architecture

In the case of the perception-based distributed architecture, a cost function considering the cost to join each cluster has been applied. We manage to group participants which are in the same local area or with the lowest possible cost into one cluster. Therefore, it is likely that a packet has to travel across less hops before reaching its cluster's leader. The parameter set for calculating the point-to-point delay in the perception-based distributed architecture is:

$$\begin{cases} \alpha_{ALM} = 3.10^{-5} \\ \beta_{ALM} = 28ms \\ d^{var}_{ALM} = 1ms \end{cases}$$

From Equ.4.49, Equ.4.51, and Equ.4.52, we can obtain:

$$d_{ALM} = (3.10^{-5} + 28).\log_k N \tag{4.52}$$

Applying $p = 1500$(bytes), we obtain:

$$d_{ALM} = 28\log_k N \tag{4.53}$$

The total delay of the perception-based distributed architectures is therefore equal to the sum of the total waiting time at all nodes (as calculated by W_{qd}) and its correspondent point-to-point delay (as calculated by d_{ALM} in Equ.4.53).

$$D_{ALM} = W_{qd} + 28\log_k N \tag{4.54}$$

4.3.3.2 Centralized architecture

In the centralized architecture, all participants have to connect to the same MCU to be served. Thus the probability that these participants are located in different countries or even different continents is very high. In such cases, the number of underlay hops that a packet has to travel in order to reach the MCU is also high. Therefore, we choose a value set which has highest value to calculate the point-to-point delay of the centralized architecture. The parameter set for calculating the point-to-point delay in the centralized architecture is:

$$\begin{cases} \alpha_{MCU} = 6.10^{-5} \\ \beta_{MCU} = 35ms \\ d^{var}_{MCU} = 20ms \end{cases}$$

From Equ.4.49, Equ.4.51, Equ.4.55, we can obtain:

$$d_{MCU} = 6.10^{-5}p + 55 \tag{4.55}$$

Applying $p = 1500$(bytes), we obtain:

$$d_{MCU} = 55 \tag{4.56}$$

The total delay of the centralized architecture is therefore equal to the sum of the total waiting time at all nodes (as calculated by W_{qc}) and its correspondent point-to-point delay (as calculated by d_{MCU} in Equ.4.56).

$$D_{MCU} = W_{qc} + 55 \tag{4.57}$$

4.3.3.3 Perception-based centralized architecture

In the perception-based centralized architecture, since only one cluster with cluster size $k = N$ is applied, the point-to-point parameter set in Equ.4.55 is applied. We obtain:

$$D_{PerMCU} = W_{qd} + 55 \tag{4.58}$$

4.3.3.4 Result Analysis

Figures 4.6, 4.7, 4.8 demonstrate the point-to-point delay among centralized, perception-based centralized and distributed architectures considering both total waiting time in queues and point-to-point transverse time at the underlay network. In the three figures, there are in average 3 enhancement video layers which are transmitted from all the participants ($n_e = 3$). We are going to analyze the results to show the effect of three main aspects on the point-to-point delay performance: (i) comparison between the distributed and the centralized architectures, (ii) impacts of the cluster size on the performance, and (iii) the newly proposed perception-based function's performance.
Regarding the comparison between the distributed architecture and the two centralized architectures, it is clear in all figures that, the point-to-point delay of the perception-based distributed architecture is far better than the centralized and perception-based centralized architectures.
Regarding the perception-based distributed architecture, the larger the cluster size (k) is, the higher the delay is. Therefore, it is better to keep the cluster size (k) small.
When the newly proposed perception-based function is applied to reduce the unnecessary traffic, the point-to-point delay performance is improved. In Figure 4.6, the base video layer and the enhancement video layers from 10% of the total number of participants are sent ($r_b = 1, r_e = 0.1$). In Figure 4.7, 50% of the peers send their enhancement video traffic ($r_b = 1, r_e = 0.5$). The last case of Figure 4.8 is when all participants send their full traffic of the base video layer and their enhancement video layers ($r_b = 1, r_e = 1$). We can see that, the more unnecessary traffic is reduced by applying our newly proposed perception-based function, the lower the point-to-point delay is. There is a performance

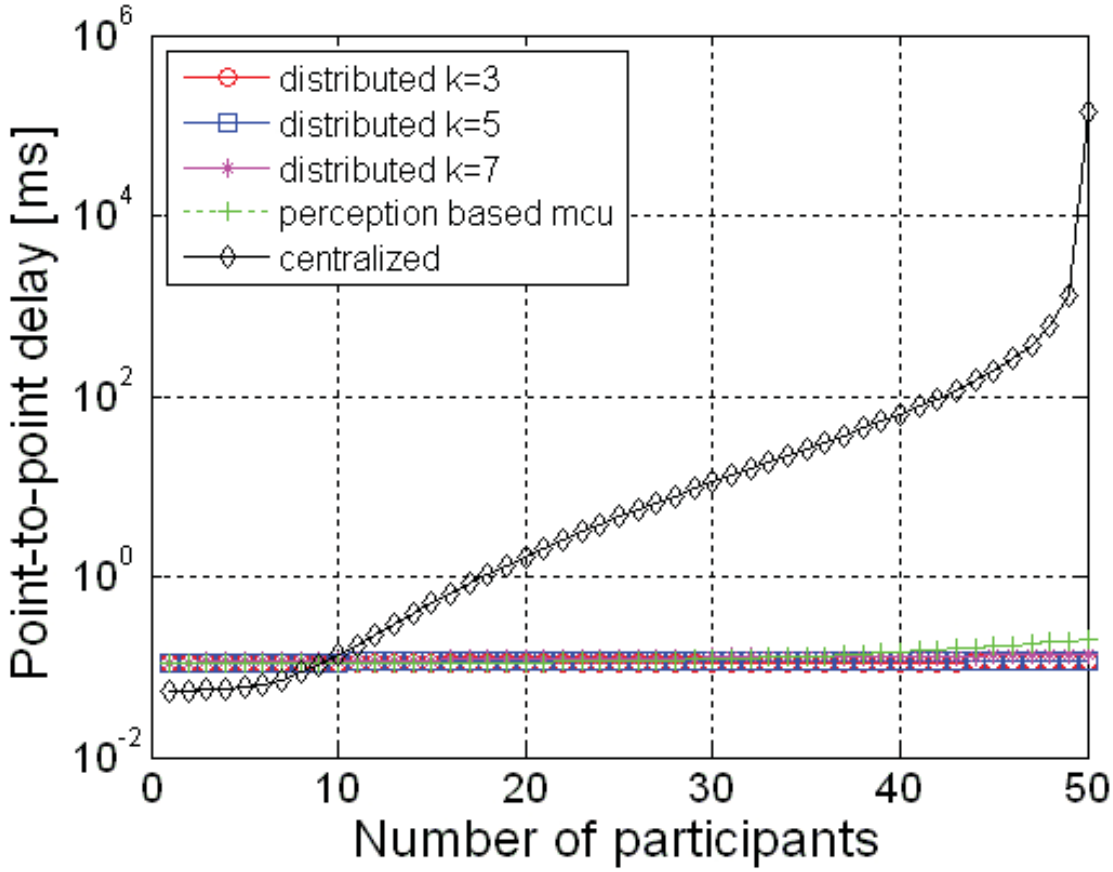


Figure 4.6: Comparison of total point-to-point delay among centralized, perception-based centralized and perception-based distributed architectures at minimum traffic when $r_b = 1, r_e = 0.1, n_e = 3$.

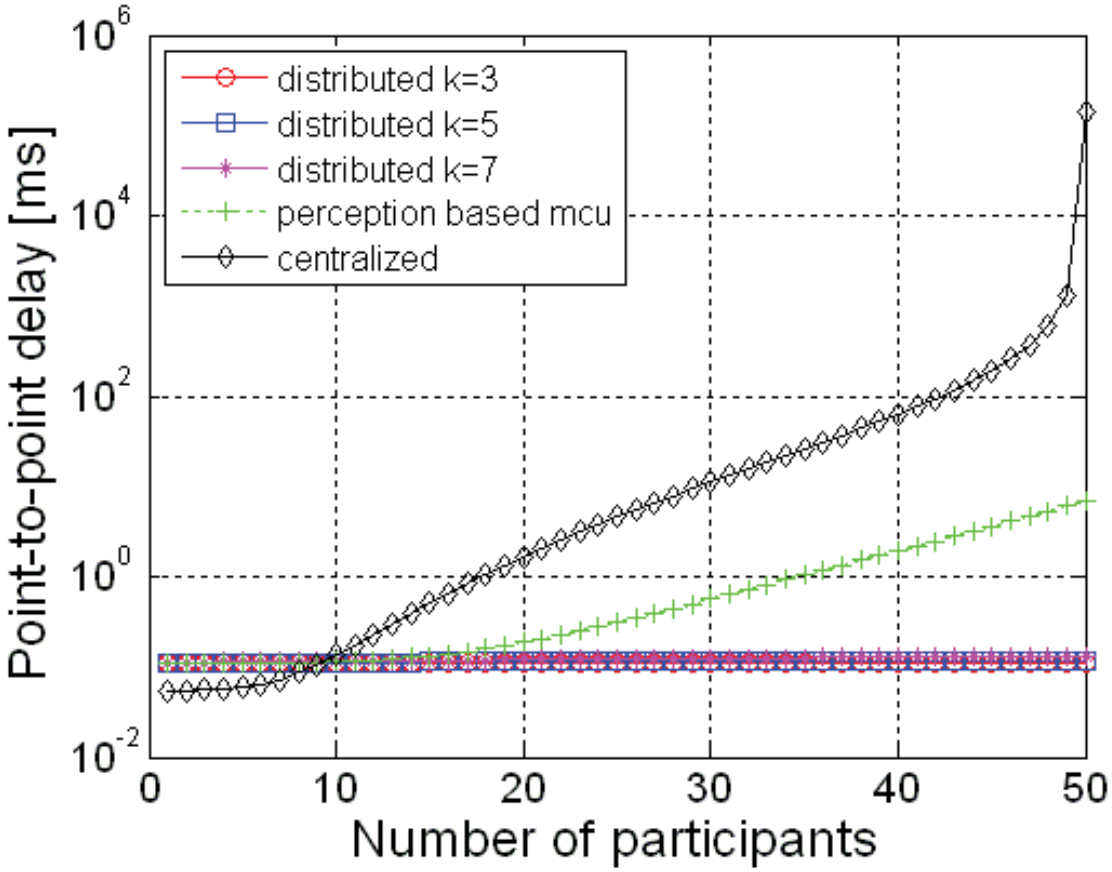


Figure 4.7: Comparison of total point-to-point delay among centralized, perception-based centralized and perception-based distributed architectures at reduced traffic when $r_b = 1, r_e = 0.5, n_e = 3$.

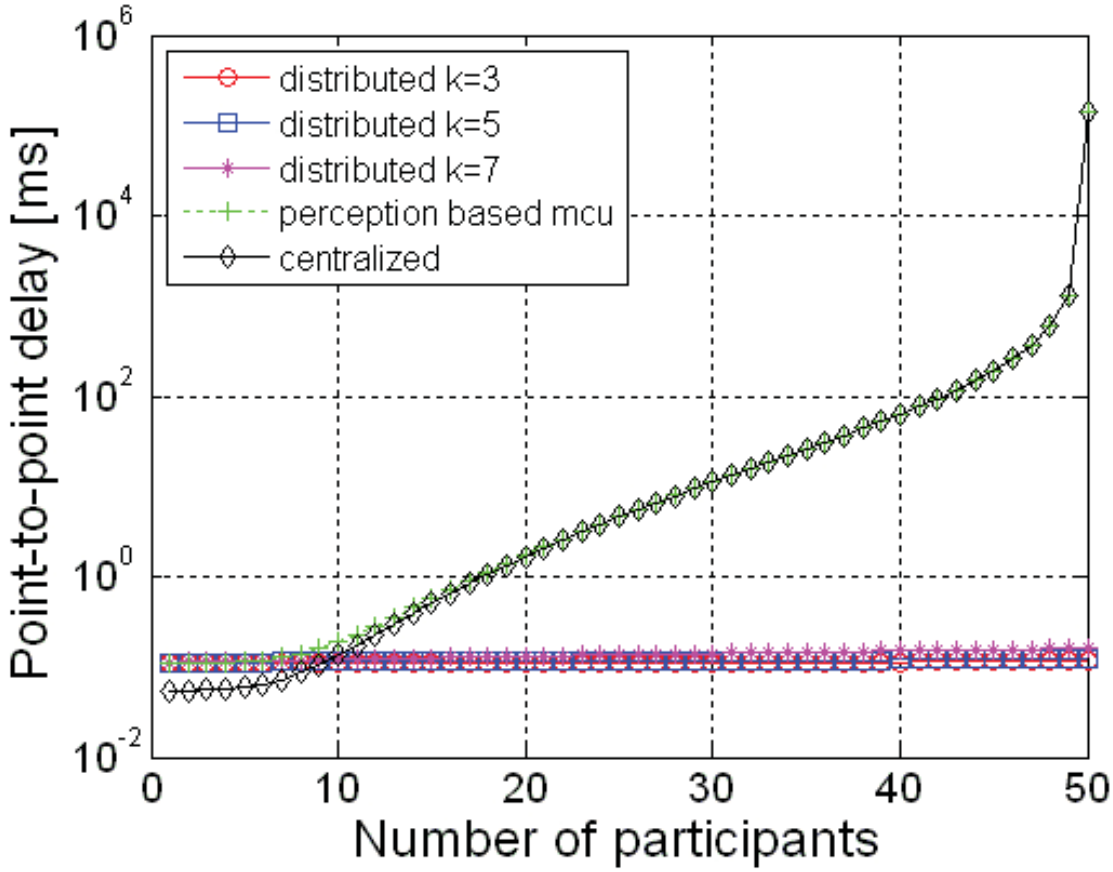


Figure 4.8: Comparison of total point-to-point delay among centralized, perception-based centralized and perception-based distributed architectures at full traffic when $r_b = 1, r_e = 1, n_e = 3$.

gain between the point-to-point delay of the centralized and perception-based centralized architectures and this gain increases when more unnecessary traffic is reduced by applying our newly proposed perception-based function.

4.3.4 Required service rate

In general, a required service rate is necessary to maintain stability in a queue. According to the queuing theory, this required service rate can be calculated by the following condition:

$$\rho = \frac{\lambda}{\mu} < 1 \tag{4.59}$$

We will calculate and analyze these requirements in details for each architecture. If an architecture can fulfill its required service rate, it can maintain the stability of the service queues.

4.3.4.1 Perception-based distributed architecture

The traffic intensity at a leader of layer l is $\rho_l = \frac{\lambda_{Al}}{\mu_l}$. In order for the queue at the leader to be in steady-state conditions, we must have $\rho_l < 1$ or $\mu_l > \lambda_{Al}$. Assuming that the perception-based leader l has been designed to support a throughput of λ_{Al} and the system can support of up to N_{max} participants, we have:

$$\begin{cases} \mu_l = M_l = (k.N_{max} + 1).\gamma^p_{hl} \\ \rho_l = \dfrac{\lambda_{Al}}{M_l} \end{cases} \tag{4.60}$$

4.3.4.2 Centralized architecture

The traffic intensity at the MCU is $\rho_m = \frac{\lambda_{AS}}{\mu_m}$ (assuming that only one server is used as the MCU). In order for the queue at the MCU to be in steady-state conditions, we must have $\rho_m < 1$ or $\mu_m > \lambda_{AS}$. Assuming that we have designed a MCU to support of up to N_{max} participants, and a throughput of λ_{AS} then the maximum throughput to be managed at the MCU is:

$$\begin{cases} \mu_m = M_m = (N^2_{max} - N_{max} + 1).\gamma^a_{hS} \\ \rho_m = \dfrac{\lambda_{AS}}{M_m} \end{cases} \tag{4.61}$$

4.3.4.3 Perception-based centralized architecture

The perception-based centralized server is designed to support a maximum of N_{max} participants. A similar requirement of the service rate is applied to maintain a steady-state:

$$\begin{cases} \mu_p = M_p = (N^2_{max} - N_{max} + 1).\gamma^p_{hp} \\ \rho_p = \dfrac{\lambda_{Ap}}{M_p} \end{cases} \tag{4.62}$$

M_p is the required service rate of the perception-based centralized architecture.

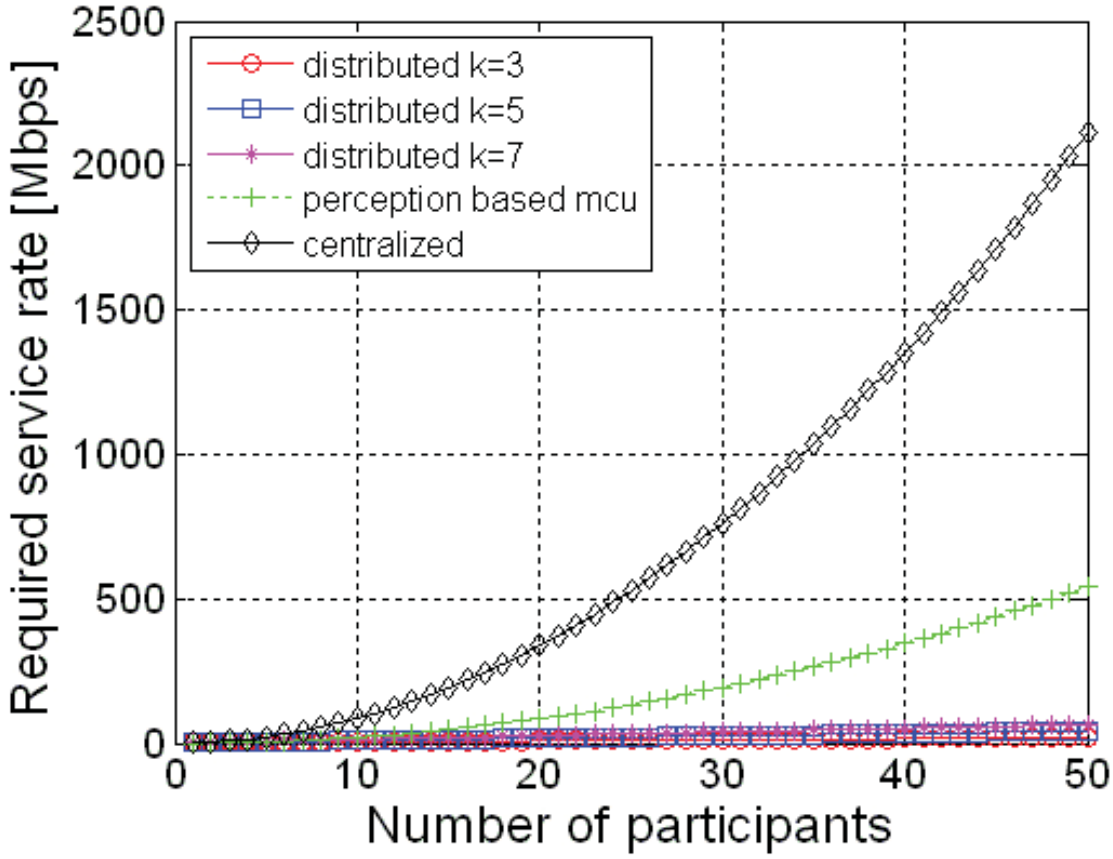


Figure 4.9: Comparison of required service rates among centralized, perception-based centralized and perception-based distributed architectures at minimum traffic when $r_b = 1, r_e = 0.1, n_e = 3$.

4.3.4.4 Result Analysis

Figures 4.9, 4.10, 4.11 show the required service rates among the centralized, perception-based centralized and distributed architectures for three different traffic configurations ($r_e = 0.1, r_e = 0.5, r_e = 1$). They are obtained from Equ.4.61, 4.60 and 4.62. These are the required service rates at the leader on layer l of the perception-based distributed architecture, the perception-based centralized server p and at the MCU, respectively, in order to maintain steady states of their queues. If a steady state is maintained in a queue, the waiting time can be high but never be infinitive meaning that all of the traffic will be definitely processed. Otherwise, if the steady state is not maintained, the queues will enter a blocked state in which no more traffic can be processed and congestion happens. We are going to analyze the results to show the effect of three main aspects on the required service rates: (i) comparison between the distributed and the centralized

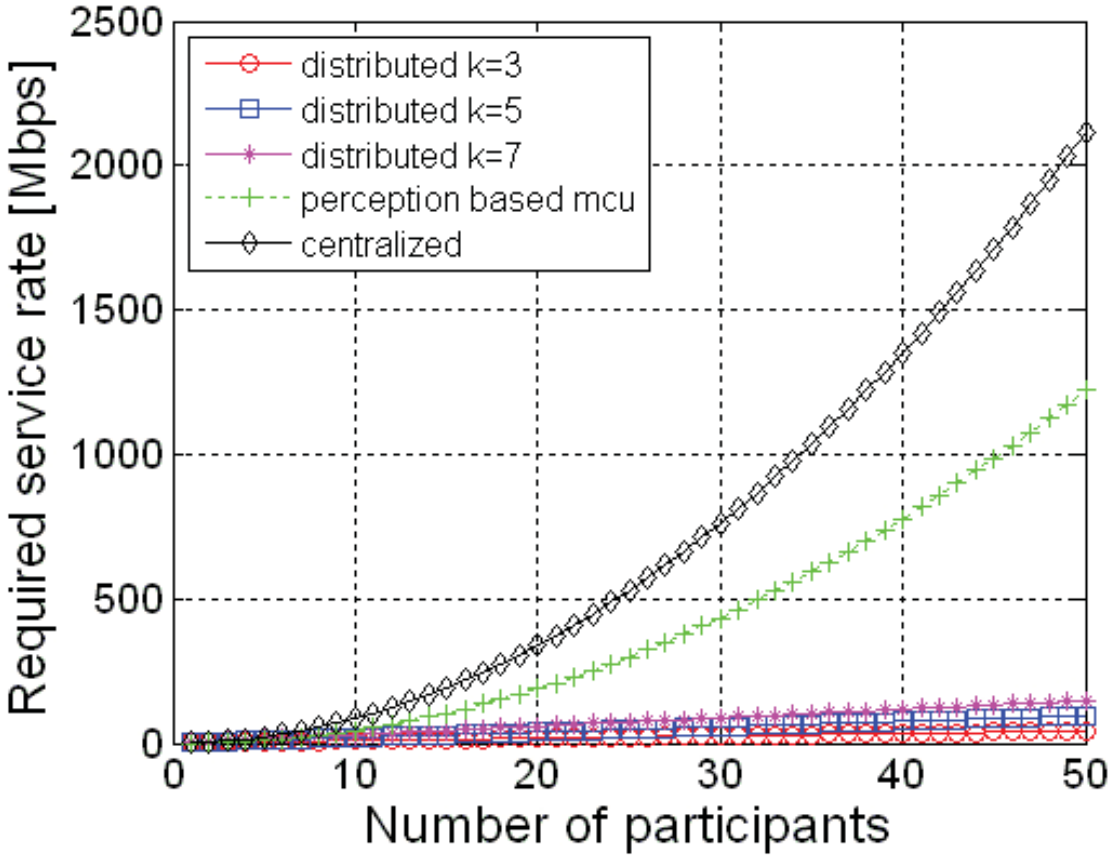


Figure 4.10: Comparison of required service rates among centralized, perception-based centralized and perception-based distributed architectures at reduced traffic when $r_b = 1, r_e = 0.5, n_e = 3$.

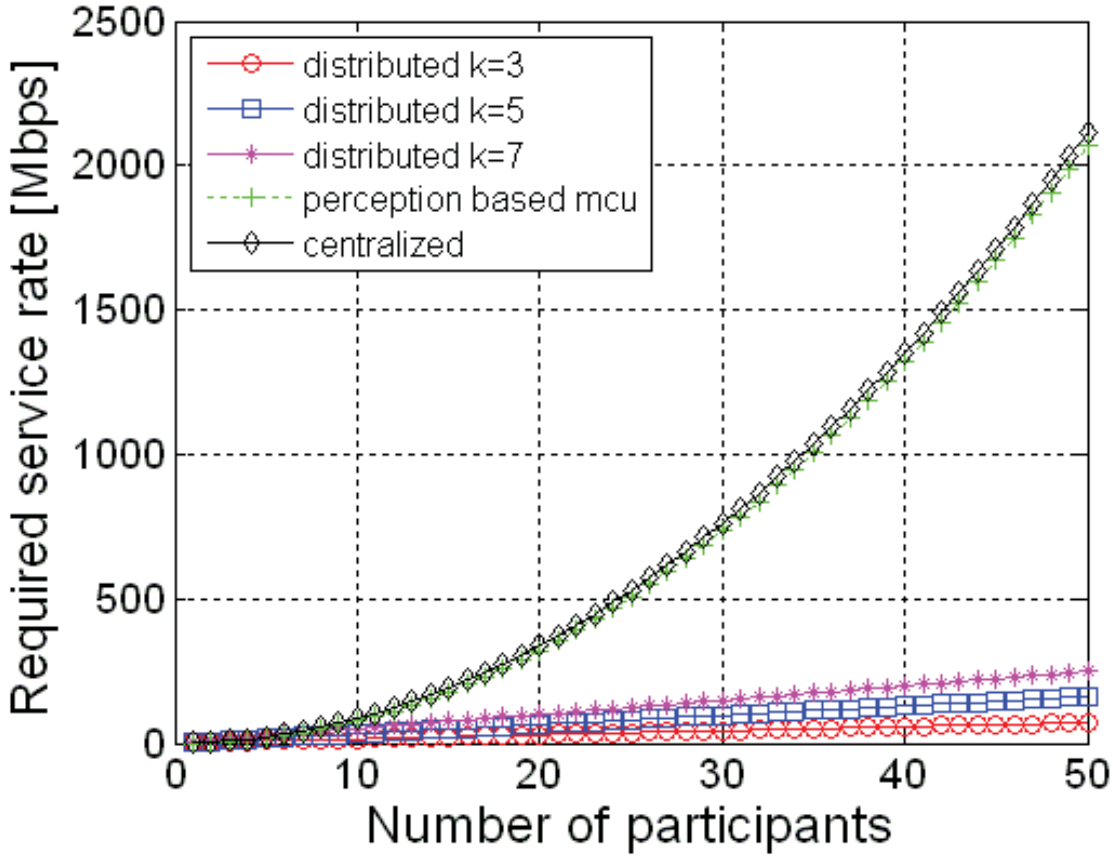


Figure 4.11: Comparison of required service rates among centralized, perception-based centralized and perception-based distributed architectures at full traffic when $r_b = 1, r_e = 1, n_e = 3$.

architectures, (ii) impacts of the cluster size on the performance, and (iii) the newly proposed perception-based function's performance.
It is clear from the three figures that the required service rate at the perception-based distributed architecture is much smaller than it is at the centralized and perception-based centralized architectures. In terms of the required service rate, the distributed architecture requires a much lower service rate than the centralized architectures.
With the same perception-based distributed architecture, and with the total number of 50 participants, the required service rate increases when the cluster size increases. Therefore, for this configuration of video conference, it is recommended to use a smaller cluster size.
All three figures show that, when the newly proposed perception-based function is applied, the perception-based centralized architecture can reduce the required service rate in comparison with the centralized architecture. The more unnecessary traffic is reduced by applying our newly proposed perception-based function, the lower the required service rate is.
There is an obvious relation between the required service rate and the price of the solution built from each architecture. This relation can be exponential. We can conclude that, the distributed architecture and the newly proposed perception-based function, when applied, can reduce the required service rate and therefore the cost of the video conferencing service.

4.3.5 Estimation of global throughput in the network

In this section, we will estimate the global throughput generated by the three architectures. The purpose is to prove that the newly proposed perception-based distributed architecture does not actually increase the total traffic transmitted on the overlay network. It can even reduce the total traffic in case the perception-base architecture is applied to throttle the enhancement video layer traffic.

4.3.5.1 Perception-based distributed architecture

Considering a peer j at layer l of the perception-based distributed architecture, we have the throughput from a peer j to a leader l and the throughput from the leader l to the peer j are $\lambda_{j\to l}$ and $\lambda_{l\to j}$, respectively.
The throughput from a peer j to a leader l ($\lambda_{j\to l}$) can be calculated by adding the traffic arriving from each participant:

$$\lambda_{j\to l} = \sum_{h=1+(j-1)k^{(l-1)}}^{jk^{l-1}} \gamma_{hl}^{p} \tag{4.63}$$

$$\lambda_{l\to j} = \sum_{h=k^{l}+1}^{N} \gamma_{hl}^{p} + \left[\sum_{h=1}^{k^{l}} \gamma_{hl}^{p} - \sum_{h=1+(j-1)k^{(l-1)}}^{jk^{(l-1)}} \gamma_{hl}^{p} \right] \tag{4.64}$$

The total throughput in a cluster is calculated by the throughput from all k participants and the multicasting throughput from the leader l to k participants.

$$T^l_{cluster} = \sum_{h=1}^{k^l} \gamma^p_h + k \sum_{h=k^l+1}^{N} \gamma^p_h + (k-1) \sum_{h=1}^{k^l} \gamma^p_h \tag{4.65}$$

$$T^l_{cluster} = k \sum_{h=1}^{N} \gamma^p_h \tag{4.66}$$

There are totally $\frac{N}{k^l}$ clusters at layer l. Thus, the total throughput at layer l is:

$$T^l = \frac{N}{k^{(l-1)}} \sum_{h=1}^{N} \gamma^p_h \tag{4.67}$$

Since there are l_{max} layers, the total throughput in the perception-based distributed architecture is:

$$T_d = \sum_{l=1}^{l_{max}} T^l = kN \sum_{h=1}^{N} \gamma^p_h \left(\sum_{l=1}^{l_{max}} \frac{1}{k^l} \right) \tag{4.68}$$

Since we can calculate the sum by $\sum_{l=1}^{l_{max}} \frac{1}{k^l} = \frac{(N-1)}{(k-1)N}$ in which $l_{max} = \log_k N$. Replace these calculations into Equ.4.68, we have:

$$T_d = \frac{k(N-1)}{k-1} \sum_{h=1}^{N} \gamma^p_h \tag{4.69}$$

4.3.5.2 Perception-based centralized architecture

In the case of a Perception-based centralized architecture, we mainly have the throughputs from participant j to the perception-based centralized leader L ($\lambda_{j \to L} = \gamma^p_{h=j}$) and vice versa from the perception-based centralized leader L to the participant j ($\lambda_{L \to j} = \sum_{h=1}^{N} \gamma^p_{hl} - \gamma^p_{h=j}$). Therefore, the total throughput in this Perception-based centralized architecture is:

$$T_p = N \sum_{h=1}^{N} \gamma^p_h \tag{4.70}$$

4.3.5.3 Centralized architecture

The total throughput of the centralized architecture can be derived from Equ.4.70 by replacing γ^p_h with γ^a_h.

$$T_m = N \sum_{h=1}^{N} \gamma^a_h \tag{4.71}$$

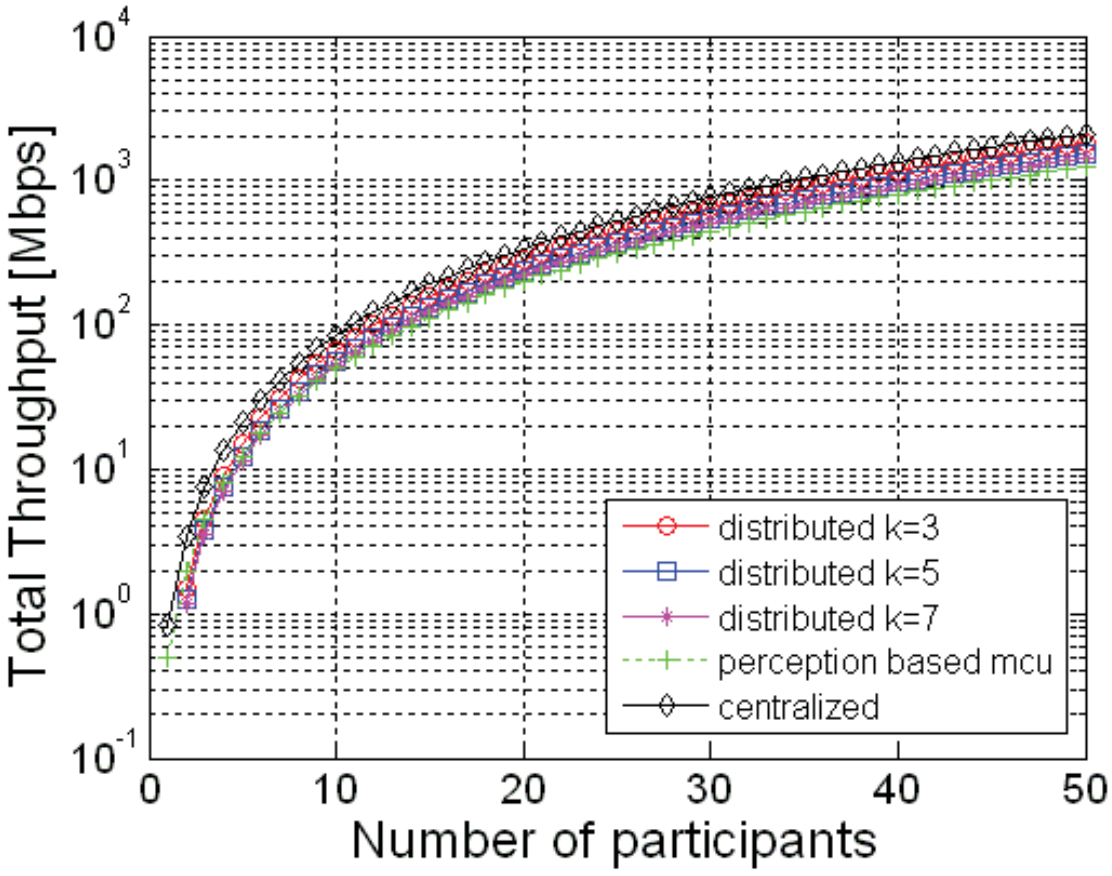


Figure 4.12: Comparison of total throughput among centralized, perception-based centralized and perception-based distributed architectures at reduced traffic when $r_b = 1, r_e = 0.5, n_e = 3$.

4.3.5.4 Result analysis

From Figures 4.12 and 4.13 we can see the total throughputs in the centralized architecture (Equ.4.71), the perception-based centralized architecture (Equ.4.70) and the perception-based distributed architecture (Equ.4.69). We find that, in fact, the total throughput of the perception-based distributed architecture is equivalent to the total throughput of both the centralized and perception-based centralized architectures when 50% ($r_e = 0.5$) and all ($r_e = 1$) participant sends their enhancement video layers. We can hardly realize the difference between these throughputs in a logarithm plot. When the newly proposed perception-based function is fully applied in Fig.4.14 so that only 10% of the participants send their enhancement video layers ($r_e = 0.1$), the total throughput of the perception-based centralized and perception-based distributed can be even far lower than the total throughput of the centralized architecture when no perception-based func-

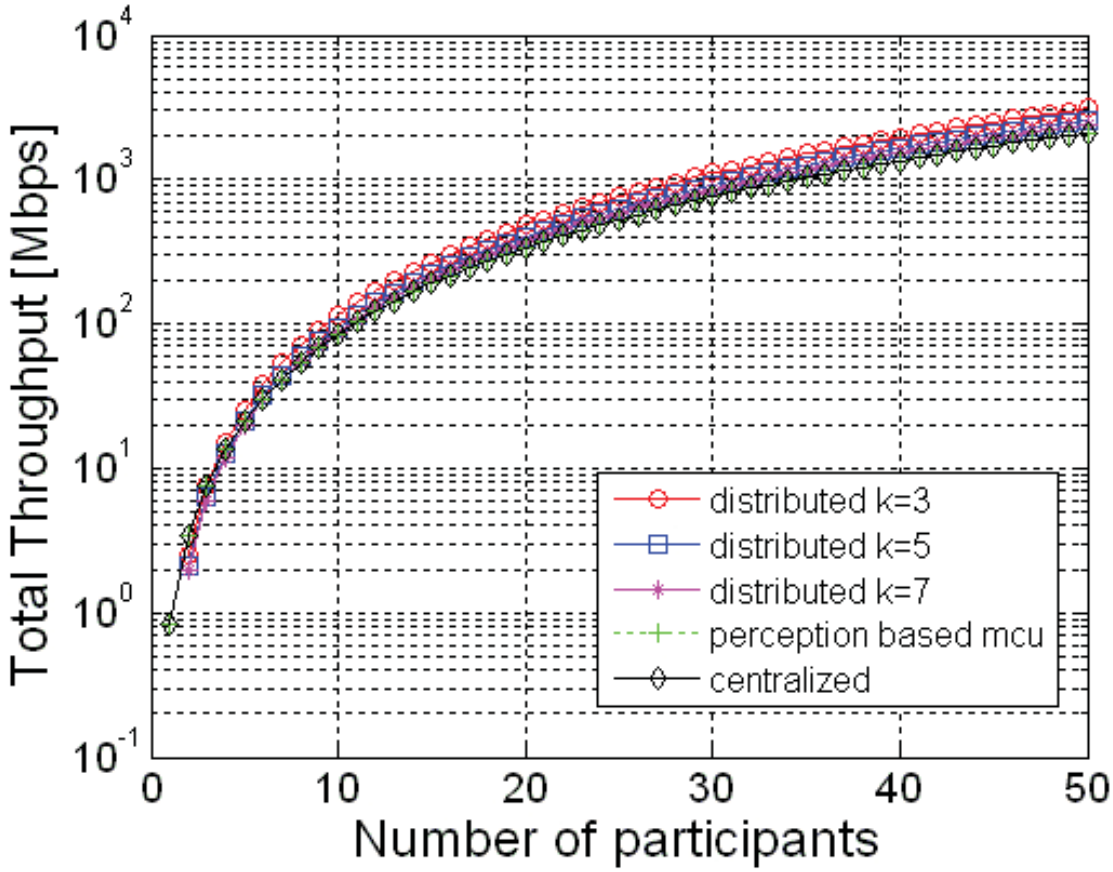


Figure 4.13: Comparison of total throughput among centralized, perception-based centralized and perception-based distributed architectures at full traffic when $r_b = 1, r_e = 1, n_e = 3$.

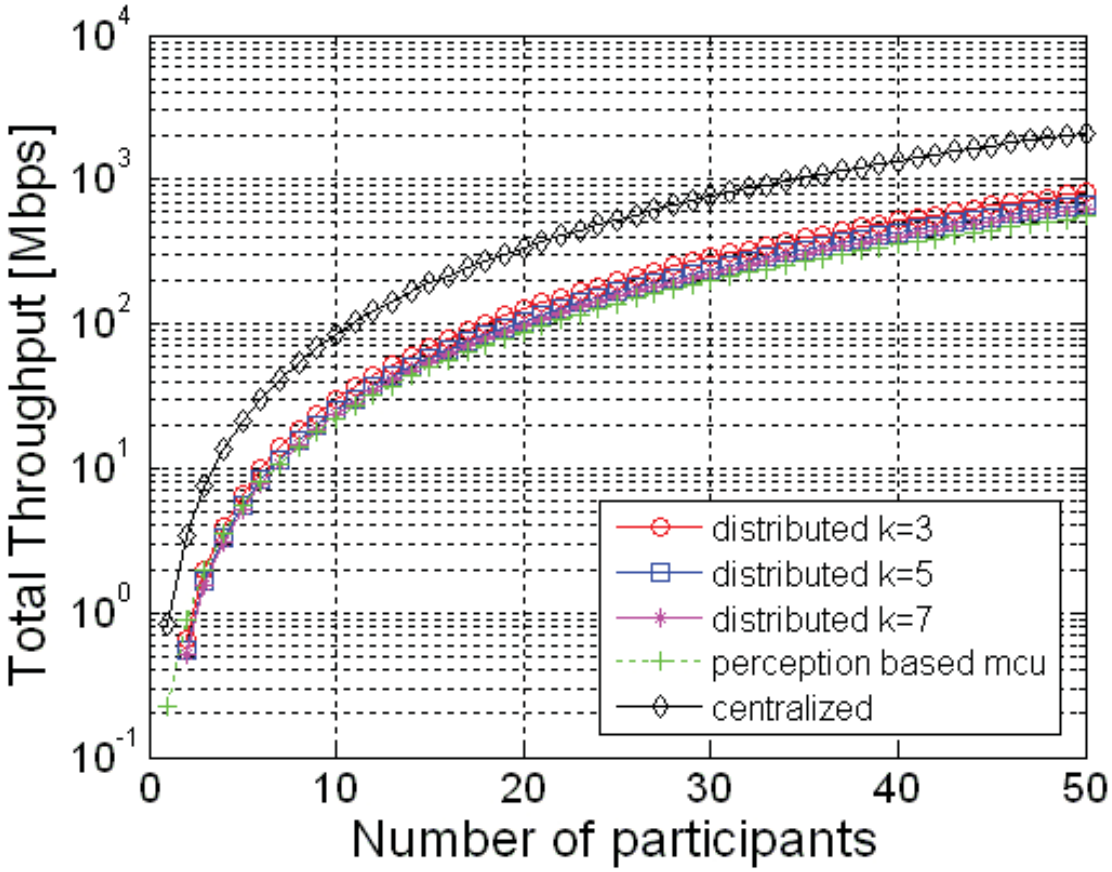


Figure 4.14: Comparison of total throughput among centralized, perception-based centralized and perception-based distributed architectures at minimum traffic when $r_b = 1, r_e = 0.1, n_e = 3$.

tion is applied.
This will answer many concerns about whether the perception-based distributed architecture has to manage a higher total throughput than the two conventional centralized architectures or not. The answer is clearly no. It can even reduce the total traffic when the perception-base architecture is applied to throttle the enhancement video layer traffic.

4.4 Conclusion

In this chapter, a new enriched distributed video conferencing architecture considering the limitation of human's perception has been proposed. Mathematical analysis and models have been built and compared for centralized, perception-based centralized and perception-based distributed architectures using queuing theory in terms of total waiting time, point-to-point delay, required service rates, and total throughput. It is worth noticing that all the enriched features of the proposed video conferencing architecture have been modelled in details and included in the mathematical analysis and expressions. The theoretical models and mathematical expressions allow us to evaluate the performance in the four different criteria for the three different architectures in real-time and with an arbitrary number of participants. It is capable of evaluating the performance of the three different architectures in real-time and with very heterogeneous contexts of peers such as peers with terminals of different screen sizes, variable bandwidth, different computational capacities and a variety of users' preferences. Numerical simulations obtained from the theoretical analysis models and the off-line statistical data have been done in the context of a multi-party multi-layer video conferencing service. The results show that the newly proposed perception-based distributed architecture can effectively reduce the total waiting time, the end-to-end delay, required service rate in comparison with the centralized and perception-based centralized architectures. Regarding the total throughput, the perception-based distributed architecture have an equivalent traffic with the centralized and perception-based centralized architectures. The results have shown a great advantage of the perception-based distributed architecture against the centralized architecture and the perception-based centralized architecture especially when the total number of conferencing participants increases.
Regarding the original question we try to solve from the beginning of the chapter for whether the distributed architecture is better than the centralized architecture or not, we can conclude that, from three different criteria (total waiting time, end-to-end delay and required service rate) that we have chosen to compare, the distributed architecture outperforms the two conventional centralized architectures. Especially when the total number of participants is large. In terms of the total throughput, it is also shown that the distributed architecture has an equivalent performance with the centralized architecture.
Our mathematical models and expressions can be used to determine the optimal cluster size for a given number of participants of the distributed video conference.

The newly proposed perception-based function when applied can reduce the total waiting time, point-to-point delay, the required service rate. It can maintain an equivalent or even can reduce the total throughput of the perception-based distributed architecture in comparison with the centralized architecture.
While the distributed architecture is better than the centralized architecture for a scalable multimedia conferencing service, it brings many problems to users who are using a wireless network to participate into the conferencing service. A special solution should be find out for mobile users in the next part of the thesis.

Chapter 5

IMS-based distributed multimedia conferencing services for Next Generation Mobile Networks

Contents

5.1 Introduction 83
5.2 3GPP IMS-based conference architecture 84
5.3 Proposed IMS-based distributed video conferencing service . . . 85
5.3.1 Inter-connectivity with LTE/WiMAX networks 85
5.3.2 Design requirements 86
5.3.3 Proposed protocol of the IMS-based LTE/WiMAX distributed conferencing service 87
5.4 Prototype and performance evaluation 93
5.4.1 Evaluation method 93
5.4.2 Evaluation results 95
5.5 Conclusion 96

5.1 Introduction

Distributed architecture offers many advantages compared to centralized architecture in terms of providing multimedia services. However, as a trade-off, distributed architecture requires that peers contribute a portion of their bandwidth and computational capacity to maintain the mutual overlay inter-connection. This requirement develops into a serious problem for mobile users and wireless infrastructure, as the radio resource in this network is tremendously expensive, and is one of the reasons why distributed architecture has not been widely applied in next generation (4G) networks. It is also the main reason why multimedia services such as video conference have to rely on a costly centralized architecture built over an expensive Media Resource Function Controllers (MRFC) via the IMS (IP Multimedia Subsystem). This chapter proposes a new distributed architecture utilizing intelligence and extra capacity, currently available on LTE and WiMAX's Base Stations to reduce the required bit-rates that each peer has to provide in order to

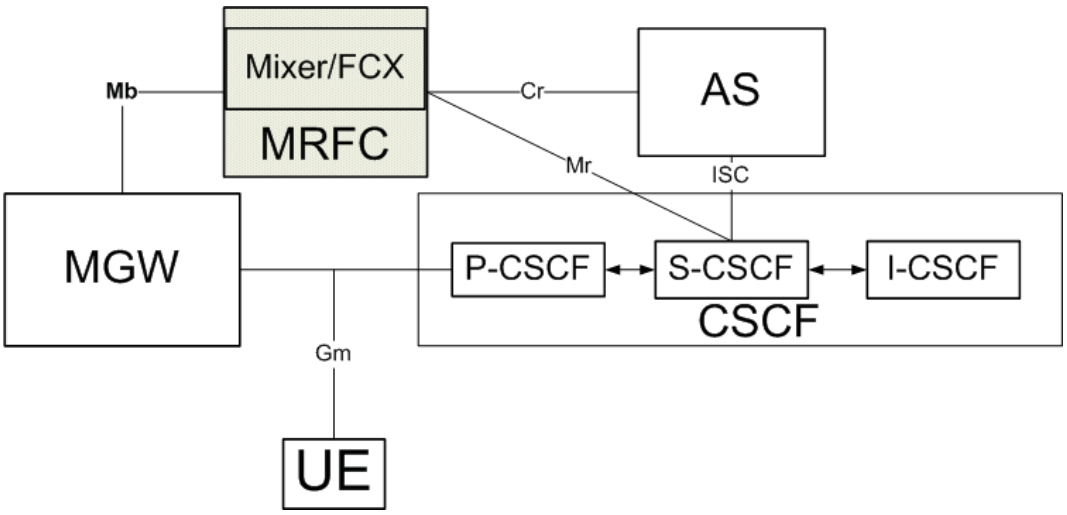


Figure 5.1: 3Gpp IMS conference architecture [92]

maintain the overlay network. This reduction saves valuable radio resources and allows a distributed architecture to provide video conferencing services on 4G networks, with all the advantages of a distributed architecture such as flexibility, scalability, smaller delay and lower cost. In addition, this can be impemented with a minimum modification of the standardized IMS platform and the 4G infrastructure, thereby saving the operators and service providers from excessive investments. A prototype has been built to prove the feasibility of the proposed architecture and evaluate its performances. The results show that our proposed distributed video conferencing service can actually reduce the average bandwidth required for data and signaling messages at wireless mobile terminals while maintaining the main operations of a video conference session.

5.2 3GPP IMS-based conference architecture

Figure 5.1 shows the 3Gpp standard architecture for the IMS-based conference architecture. Here Session Initiation Protocol (SIP) and Real-time Transfer Protocol (RTP) are used as the main signaling and media transportation protocols. The Call Session Control Functions (CSCFs) are entities that route SIP messages. The media gateway (MGW) is the entity that handles/forwards RTP traffic down to the UE when necessary. The conference focus is in both the Media Resource Function Controller (MRFC) and in the conferencing AS. The MRFC provides all of the media related functions (e.g., mixing, transcoding, trans-rating...) required for conferencing. It may also contain a Floor Control Server (FCS) function. Floor control is the term for managing the system according to user status (in/out/pause/return) and conference status (active users, chairman). Since all of the media related functions for conferencing are done in the MRFC, this architecture is highly centralized. To overcome the disadvantages of the centralized

architecture, a number of concepts have been proposed to support distributed conferencing service architecture on the current centralized architecture of IMS-based conferencing services on LTE/WiMAX networks. One example proposes that the FCS feature is proposed to be separated from the MRFP [93]. That modification, however, that does not change the centralized nature of the architecture. In another effort [94], a distributed solution is proposed as an overlay network of centralized conferencing clouds. However, that architecture would not provide for a proper integration with any specific overlay algorithm. Moreover, even-though a Content Distributed Network (CDN) with proxy servers has been constructed to support this integration, clients still have to process all of the signaling and media loads. The proxy servers mainly serve as proxy MCUs to connect several clients together using a centralized architecture and then connect all of those centralized groups together by creating an overlay of proxy servers.
To conclude, the conventional methods for IMS-based video conferencing services are either centralized (in one form or another) or they have not fully utilized the capacity of the 4G infrastructure.

5.3 Proposed IMS-based distributed video conferencing service

As a many-to-many communication mechanism, distributed video conferencing services are mainly built on multicast systems. IP-multicast [21] is the most efficient type of multicast today. However, its problems with deployment are preventing it from being widely applied [22]. The Application Layer Multicast (ALM) infrastructure is a promising alternative. Many ALM algorithms have been proposed and distributed video conferencing services have been built using ALM [38].
The problem is that, since ALM algorithms work on the application layer, there is no preference as to what kind of access network is used by the session terminals when they participate in a conference. In fact, many participants use a radio access network (such as LTE or WiMAX [95]) to participate in conference sessions. Thus, limited and expensive resources of the mobile terminals and of radio channels are sometimes unnecessarily expended by ALM's required operations such as heart-beating and data forwarding. While distributed conferencing service architecture can overcome many of the technical limitations of centralized architecture, the business model of the distributed conferencing architecture can create a win-win service for participants in which they can contribute their computation and get a free service in return (or they can even contribute their computation for their direct financial gain). At the same time, the distributed architecture can still support the existing business model that is provided by the centralized architecture. Our proposal to overcome these limitations has been partly presented in [96].

5.3.1 Inter-connectivity with LTE/WiMAX networks

A LTE/WiMAX system applies a ring-topology [97] where components connect together using the same core network. All eNodeBs in LTE/WiMAX systems are smart Base Station Systems (BSS) built with built-in intelligence and are capable of contributing computational capacity to the service [98]. If these eNodeB/xBS can represent UEs in handling ALM data traffic forwarding and control message processing, UEs can participate in the distributed conference as if they are participating in a conventional IMS-based centralized conference. The possibility of deploying our proposal onto a LTE/WiMAX infrastructure is discussed in this research work [99].
WiMAX is a broadband wireless access (BWA) technology for wireless metropolitan area networks. It has been fostered by the WiMAX Forum, an international industrial organization founded in June 2001 to promote the adoption of WiMAX compatible products and services [100]. A WiMAX network usually contains the following network entities [100]: an ASN (Access Service Network), a CSN (Connectivity Service Network), an ASP (Access Service Provider) and an MS/SS (Mobile station/Subscriber station). The ASN provides radio access to WiMAX subscribers, and its features and roles include: transferring of AAA authentication messages, authorization and session accounting for subscriber sessions, and radio resource management [100]. The CSN provides IP connectivity services to WiMAX subscribers meaning that the CSN covers several functions such as: Internet access, AAA proxy and server, and Policy and Admission Control based on user's subscription profiles [100]. The ASP is a business entity which provides applications or services [100]. In this entity, the WiMAX Forum proposes two types of connection to an application: via a non-IMS application server and via P-CSCF (IMS). However, it is not clear how to develop such a non-IMS Application Server [100]. Thus, we suppose that the WiMAX Forum method uses the IMS-based Application Server to provide services for WiMAX subscribers. WiMAX has two types of inter-connectivities with other wireless networks such as LTE (3GPP): loose couple and tight couple [101]. Figure 5.2 shows how loose couple inter-working utilizes the AAA-server of a 3GPP network. Data streams are not passed through the core network of a 3GPP LTE system. This method guarantees the independence of WiMAX network, however, it results in high handover latency between the two networks. The handover between WLAN and UMTS was studied [102], and the average handover latency results for loose couple and tight couple were found to be 400ms and 150ms, respectively. Therefore, this option is not suitable for real-time services. On the other hand, Figure 5.3 shows that a tight couple does apply to a RNC (Radio Network Controller) and to the core network (SGSN and GGSN). The WiMAX's BS connects to WCDMA's RNC or SGSN directly. The advantage of this mode is that it reduces the handover latency and guarantees a seamless handover. In fact, it is possible to provide real-time services for WiMAX subscribers via IMS and the tight couple connectivity with LTE. Figure 5.2 and Fig.5.3 also show the proposed blocks of xBS (in WiMAX), XeNodeB (in LTE), and xAS, and their positions inside the network. These blocks are essential in order to provide our distributed scalable video conferencing services on WiMAX/LTE networks.

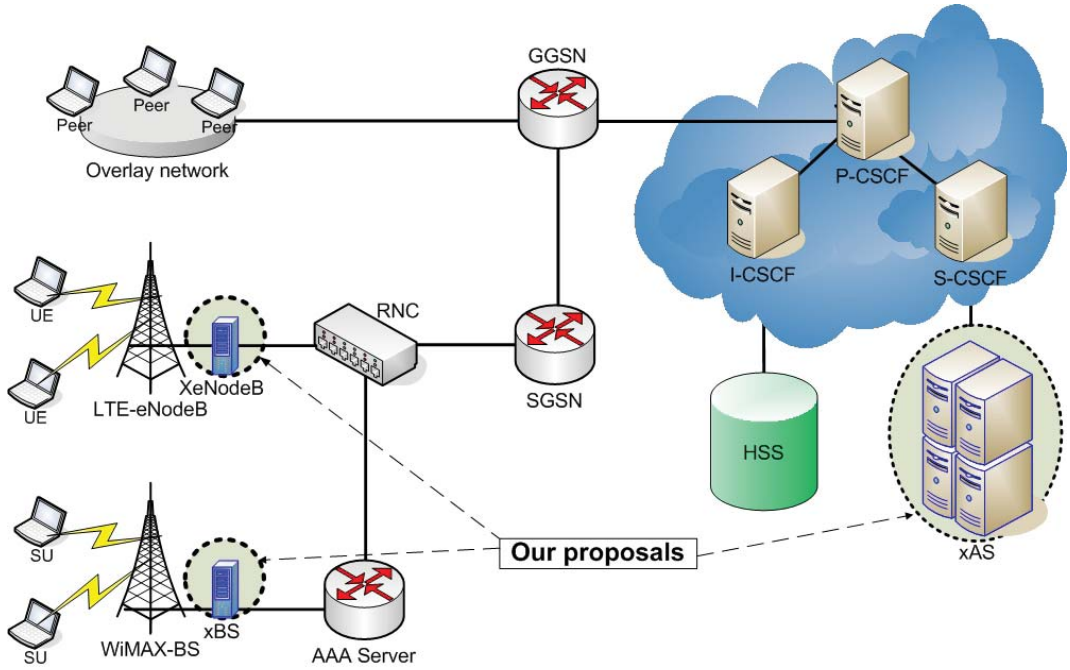


Figure 5.2: Loose couple of inter-working between LTE/WiMAX and IMS.

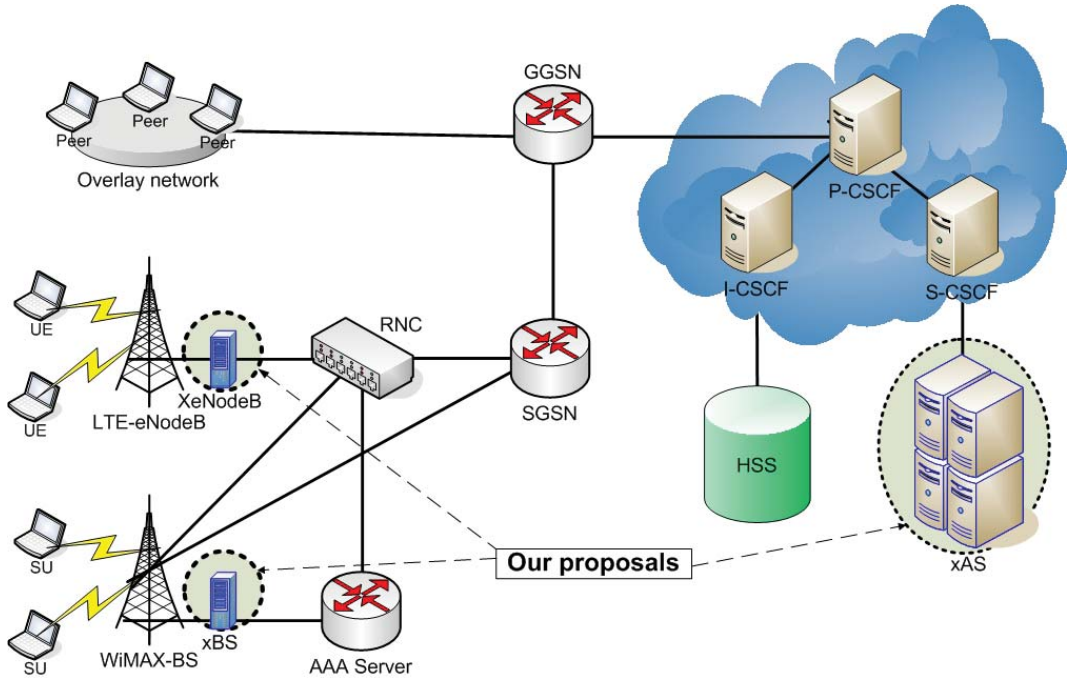


Figure 5.3: Tight couple of inter-working between LTE/WiMAX and IMS.

5.3.2 Design requirements

The main target of this section is to provide the 4G mobile terminals the capability of using a distributed video conferencing service based on the ALM-based overlay network to save the cost of expensive MCU use as well as reducing the redundant signaling and data forwarding required by that overlay network. The solution is built based on IMS's standards and 4G infrastructure. The main requirements of the design are to:

- Utilize available resource and information which can be easily obtained from the LTE/WiMAX infrastructure to acknowledge the ALM-based distributed conference so that the limited resources of the mobile terminals are used efficiently,
- Discard the standard centralized architecture that uses a MRFC thereby reducing the total expense of the entire solution and avoid the single point of failure while still maintaining multi-party conferencing features,
- Utilize floor control, a mechanism which coordinates simultaneous access to shared resources in multimedia conferences. Floor control allows applications and users to gain safe and mutually exclusive or non-exclusive access to the shared resources. Floor control can be used to avoid or resolve conflicts among simultaneous media inputs. For example, at any given time, the moderator of a floor can ensure that only one person is heard by other participants, or that one person types (writes) into a shared document. In our proposal, floor control is defined as the mechanisms with which to manage a conference session such as Join/Leave, Pause/Return, and Application layer Handover [103]. Floor control is required to provide a well-managed conference service, floor control is required. Since floor control is not yet been considered for a distributed architecture, we need to find a solution to apply floor control mechanisms into our ALM-based conferencing architecture,
- Support a seamless integration among the LTE/WiMAX mobile terminals and the ALM-based conferencing platform during a mobile video conferencing session such as join/leave, pause/return, soft handover, heartbeat, etc.; and
- Provide a QoS-guaranteed mechanism for QoS-required ALM conferencing architecture systems.

5.3.3 Proposed protocol of the IMS-based LTE/WiMAX distributed conferencing service

5.3.3.1 New features of our proposal

In this proposal, eNodeB(s)/xBS(s) are used as the proxy servers for bridging between the participating UEs and the distributed conference. It will represent the UE in the overlay network as a representative overlay node. An AS (Application Server) will be used primarily to manage the floor control and the mapping between UEs and their proxy eNodeB/xBS. We call the extended version of eNodeB that supports the ALM's

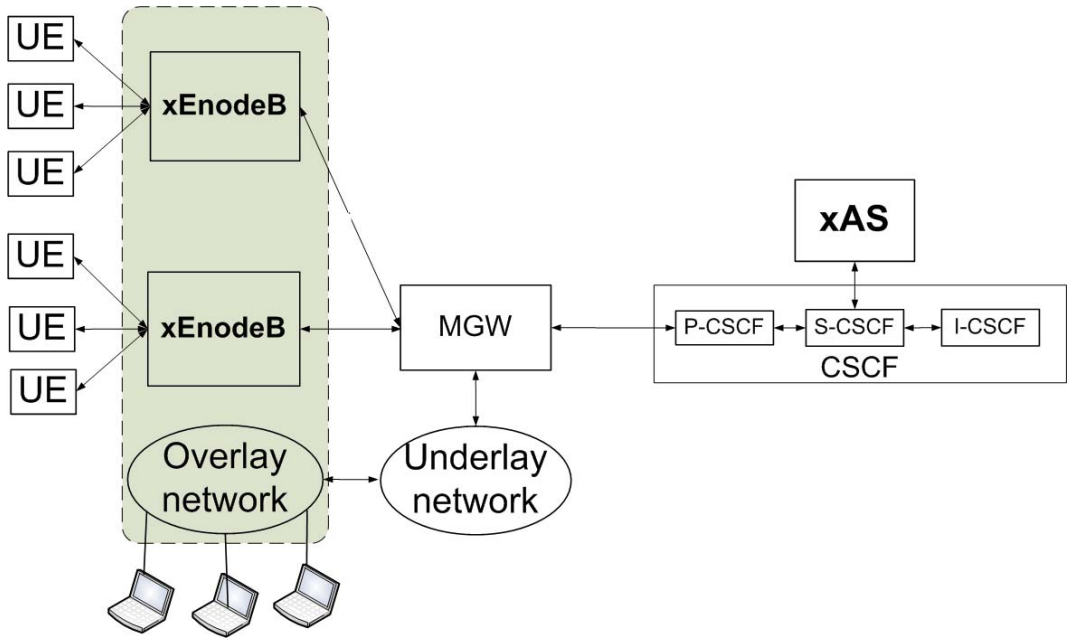


Figure 5.4: LTE/WiMAX IMS-based distributed conference.

protocol the XeNodeB, and the extended version of the AS the xAS. To obtain these design requirements (described in section 5.3.2), we propose new solutions to support a new version of eNodeB/xBS:

- **Application level soft handover**: since our proposal is based on an overlay network in which the peers are arranged in layers and clusters according to their connection conditions (bandwidth, delay, packet-loss, computational capacity, availability), it is mandatory to update this information whenever it is changed during the communication session. The xBS(s)/eNodeB(s) are used as proxies for wireless terminals to participate in the over-lay network, and so when the wireless terminals change their base stations during the handover process at the wireless layer, the handover process must also be performed at the handover process at the overlay network (application level). This is required because each base station has its own capacity and available bandwidth and each wireless terminal has to update their resource reservation process before attaching to another base station. Since the new base station will represent the wireless terminal in the overlay network, the new connection conditions must be updated to the overlay network before the connection can be actually released in the radio layer to finish the handover process. If the old overlay node is deleted before the handover process finishes, all forwarding traffic going through that node will be discarded. Therefore, for a smooth and effective soft handover process in the overlay network (application level), it is necessary to have a duplicated node in the overlay until the handover process at the radio layer is finished. The application layer handover can also reduce the

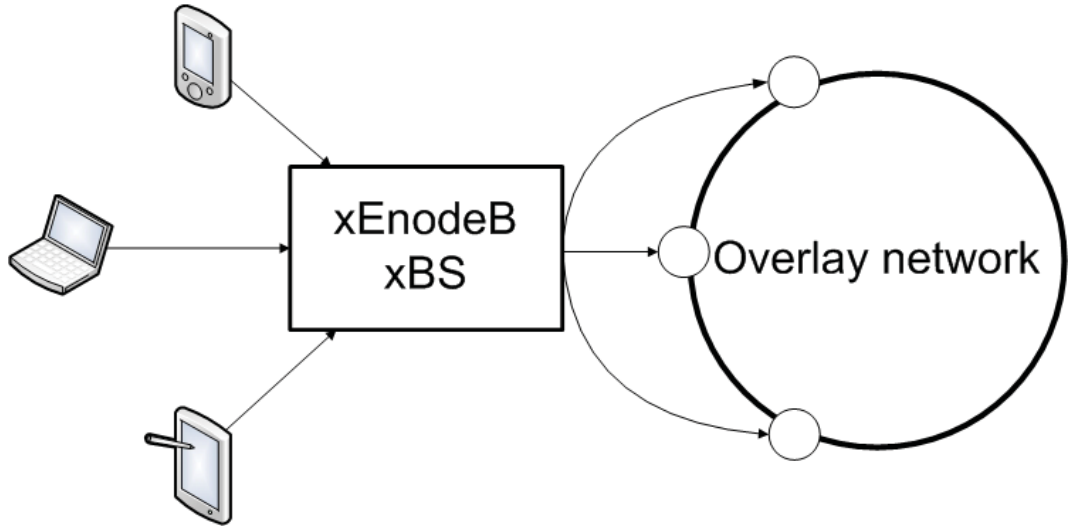


Figure 5.5: Representing nodes during Join/Leave.

unnecessary cost to the mobile subscribers that occurs during pause/return,

- **Heartbeat**: A mechanism widely applied in overlay/distributed architecture in which peers periodically send short messages to inform other peers about their existence (whether the node is still alive and connecting) in the overlay. It also receives heartbeats from other peers updating about their existence. Many ALM algorithms have to depend on a heartbeat mechanism to maintain their group, and if the UE has to directly respond to all heartbeats, it will soon run out of power and computational capacity. The heartbeat can be handled at the representative xBS/eNodeB level to prevent the UEs from being resource abused,

- **Pause-Return**: During the Pause-Return process of the UE(s) from the conference session, the representative eNodeB/xBS can do the data forwarding work for the UE(s) in the overlay network. This new feature saves considerable valuable radio resources on the part of the wireless mobile terminals,

- **Join-Leave**: While joining/leaving a conference, each UE has to pass through a QoS resource reservation process. The QoS parameters (bandwidth, delay, packet-loss) of the wireless connection is thereby confirmed for each UE. The representative eNodeB/xBS will create a node in the overlay for each UE that it represents. The position of that newly created overlay node in the media distribution tree is determined by the application-aware cost function based on the confirmed QoS resource reservation of each UE. Hence, the representative eNodeB/xBS can create several nodes in the overlay network according to the number of UEs it is representing as shown in Fig.5.5. Our proposal will enable this process with the extended version of the xBS/eNodeB. The proposal has been briefly introduced in [99].

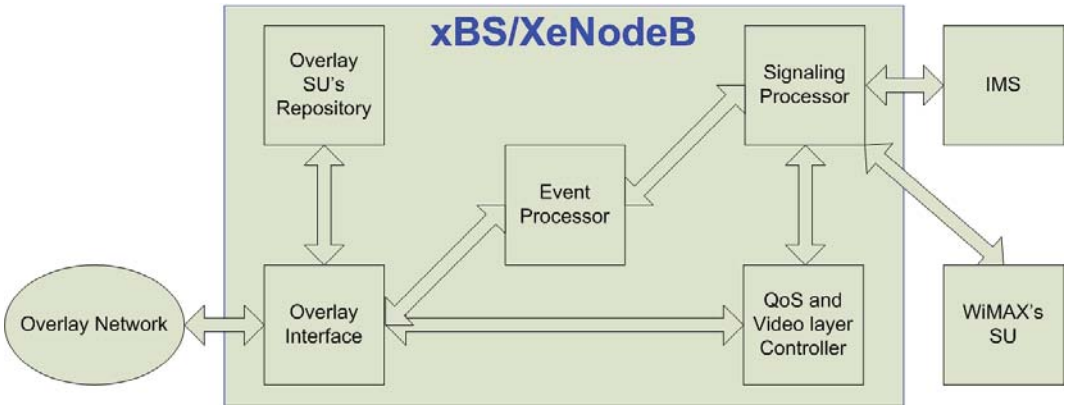


Figure 5.6: Extended features of the XeNodeB.

Many questions have been posed about whether the eNodeB(s)/xBS(s) have enough computational capacity and intelligence to enable the functionalities required in a distributed architecture? The feasibility of using eNodeB/xBS for advanced features has been investigated in [104] and [105].

5.3.3.2 Detailed descriptions of XeNodedB and xAS

Figure 5.6 shows the extended features of the **XeNodeB/xBS**:

- Participate as a node in the Overlayâs network;
- Transfer or forward data and control messages; and
- Scalable Video layer registration.

To achieve these features, each XeNodeB manages:

- Routing tables:
  - A listing UE participants served by that XeNodeB. The list contains the UEs' IP, Overlay role (source/relay/forwarding), status (idle/active), and registered video layer(s) (base/enhanced);
  - Tables are updated based on the status(idle/left/active) of their managing UEs.
- A list of peers in the distributed conference containing the peer's IP, distance (cost to reach that peer from the current XeNodeB).
- An Event Processor:
  - Updates the participant list in join/leave, pause/return and handover operations, and

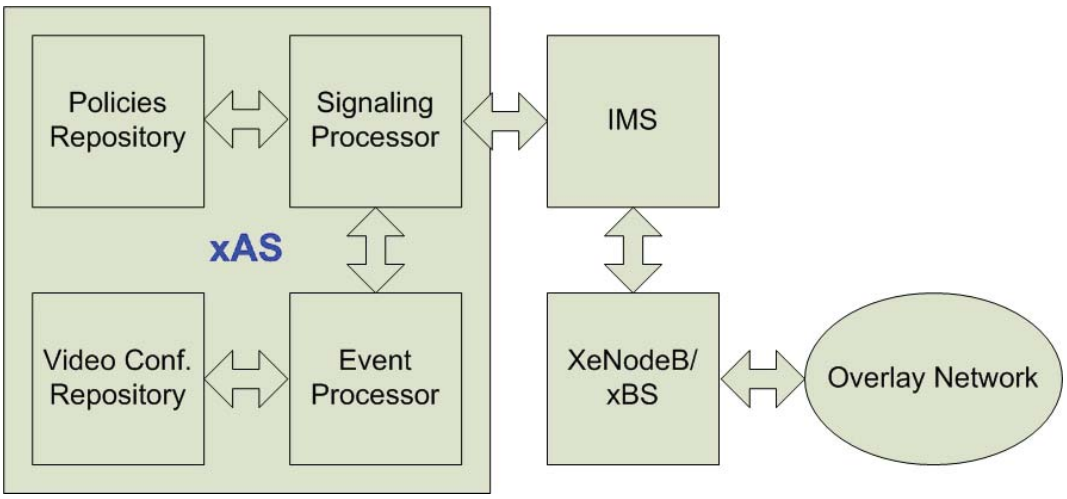


Figure 5.7: Extended features of the xAS.

  - Reports to xAS regarding the status (idle/left/active) of its managing UEs.

- An Overlay interface:
  - The interface between its participating UEs and the overlay network,and
  - Filters forwarding packets and send them back to the overlay network to save UEâs capacity.
- A Signal processor:
  - Obtains UEsâ status (availability, its available QoS level) from the xAS,
  - Sends back UEâs status to the distributed conference via the overlay interface as required.
- The QoS Video Layer registration: Checks each UEâs available QoS and its registered video layers to see if they can be matched together.

Figure 5.7 shows the extended features of the **xAS**:

- Floor control for the video conferencing application: conference ID, participant's list, participant's status,
- Obtain UE-related information from the LTE/WiMAX network and forward to the overlaying nodes,
- QoS guarantee, conference's QoS policies.

This xAS can be developed by operators and service providers and placed inside of a network. It then provides distributed video conferencing service for 4G users. xAS manages the service and operators are usually included in providing such service. To achieve these features, each xAS manages a:

- Conference list: Contains conference ID, IP, XeNode ID, Conference status (in progress/terminated),
- Signaling processor:
  - Receives requests from XeNodeB's Signaling Processor; and
  - Interrogates the UEâs (QoS, availability...) from the Home Subscriber Server (HSS) and IMS and sends it back to XeNodeB's Signaling Processor,
- Event processor:
  - Receives updates from XeNodeB's Event Processor about leave/join, pause/return, and handover; and
  - Updates the Conference list,
- Conference policy, including the:
  - Starting time, duration, maximum number of participants,
  - QoS requirements, and
  - Billing information.

Figure 5.8 shows the call flow when a UE wants to participate in or leave a distributed video conference. Firstly, when turned on, it automatically sends a REGISTER message to its XeNodeB. The XeNodeB then updates its routing table and sends $(eNodeB - ID, UE - SIP - Account)$ to the xAS. This information will be stored in the xAS's Conference List. When the UE wants to initiates its participation in a distributed conference, it sends the INVITE(Conference ID, Layer Registration) message, containing the maximum number of enhancement layer(s) it wants to receive from the conference multicast tree, to the controlling XeNodeB. The XeNodeB converts the layer(s) number to a QoS parameter which is comprehensible to the Policy Charging Rule Function (PCRF). Next, the INVITE message is forwarded to the xAS for mapping among the Conference-ID, UE-SIP-Account, and XeNodeB-ID in the xAS's conference list. At the same time, the INVITE message is forwarded to the Policy Charging Enforcement Function (PCEF) using the Authorization-Authentication Request (AAR) message sent by the Call Session Control Function (CSCF). The PCRF will check whether the UE has subscribed for enough QoS resource in order to receive the required number of enhancement layers. If its subscription is sufficient, a resource reservation request will be sent to the PCEF to activate the resource policy for the UE to join the distributed conference [106]. The confirm resource is then reported to the xAS for updating of the QoS requirements and of the billing information in the Conference Policy [105]. Upon receiving the QoS confirmation, the XeNodeB sends an ALM's JOIN-REQ(UE-SIP-Account, Layer-Reg) message to the ALM group to represent the UE participating in the ALM tree. After a new node has been successfully added to the ALM tree, a JOIN-REP(UE-ID, UE-SIP-Account, Lay-Reg) is sent to the XeNodeB. The UE-ID is assigned by the ALM

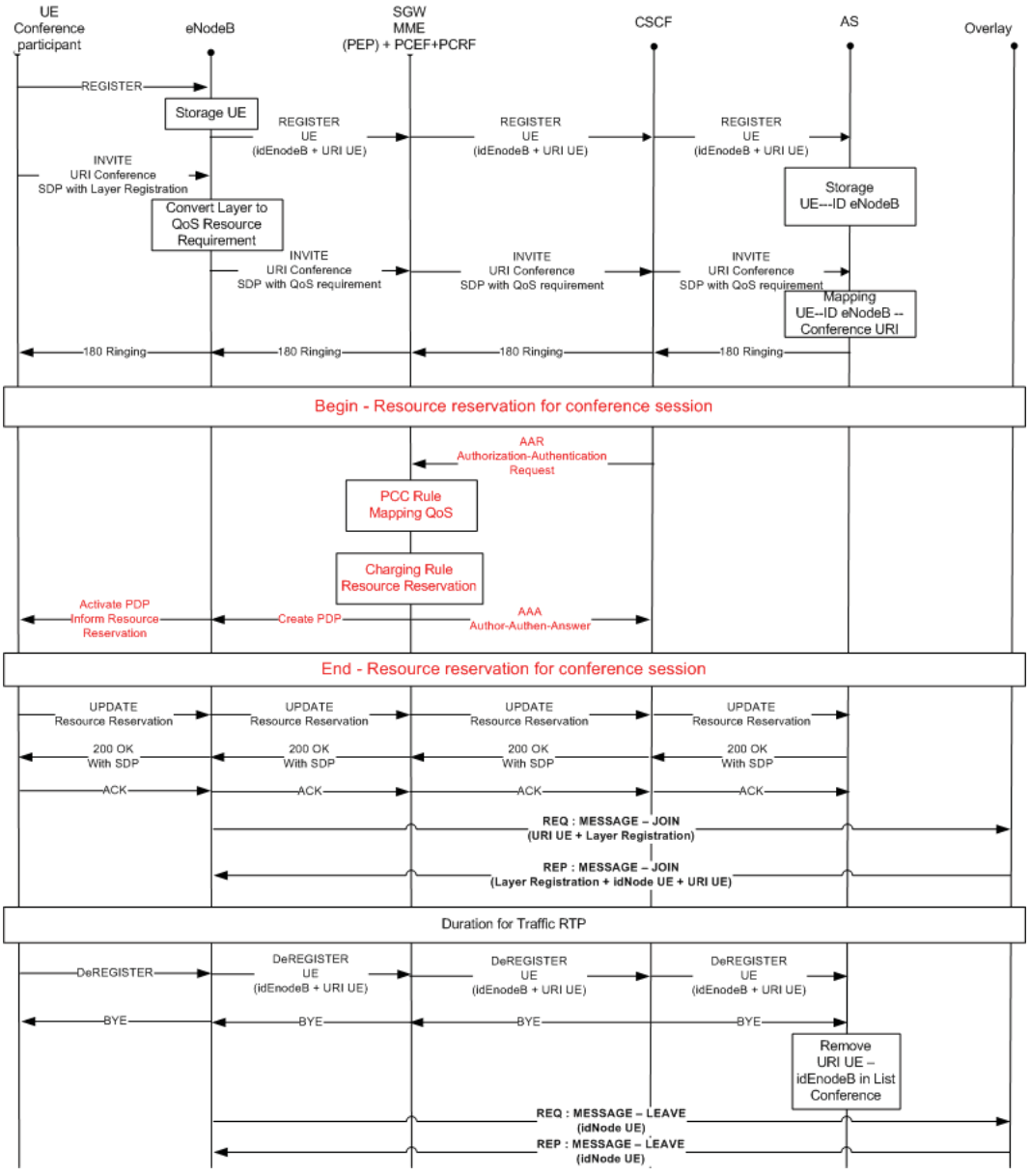


Figure 5.8: The UE Join/Leave process to/from the overlay-based distributed conference with QoS support.

and reported to the XeNodeB for management purposes. When the UE wants to leave the conference, a De-REGISTER(eNodeB-ID, UE-SIP-Account) is sent to the xAS and then a REQ-LEAVE(UE-ID) is sent to the ALM tree for a leaving request. The UE's record is then removed from the xAS's Conference List.
Figure 5.9 illustrates the Pause/Return operations of a UE over a distributed video conferencing service. During the conference session, the UE may want to suspend the service (it does not want to receive or transmit video streams and signaling messages for a time duration) while preferring to be able to return to the ALM group with the smallest delay afterwards. Of course, if the suspending time is too long, the UE will be automatically discarded from the ALM group. When a UE wants to become idle, it sends the PAUSE(UE-SIP-Account) to the XeNodeB and to the xAS so that they can update their lists. The XeNodeB then automatically sends a PAUSE-REQ(UE-ID) to the ALM group. The ALM group will stop sending bit-streams and signaling messages (HEARTBEAT) to the specific UE-ID, thus placing that UE on the waiting list of the ALM group within a certain time period. If the UE returns to the conference within that time period, it will only have to send a RETURN(UE-SIP-Account, UE-ID) to the RDV point of the ALM group and then comes back to the ALM tree. If the UE is idle beyond the set time period, it will automatically be discarded from the ALM group's waiting list. When working in the idle mode, the UE returns its reserved resources to the network via a Resource Modification process. The resource will be given back to the UE when it returns.
Figure 5.10 shows a UE going through a handover process while participating in a distributed conference. This process is compatible with the 3GPP recommendation on the radio layer handover process in LTE/WiMAX [107]. The radio layer Handover process starts with the hard handover of the mobile terminal in the radio level between two eNodeB/xBS [108]. It starts when the original eNodeB finds that the radio connection with the UE is reducing to below a pre-defined level. The original eNodeB will contact the destination eNodeB with an HO Request. If everything is ok, the destination eNodeB will acknowledge with a HO Response. The original eNodeB notifies the UE that it can start the handover process. At the service layer, when a UE is about to be handed over from the source to the destination XeNodeB, the destination eNodeB sends an APPLICATION-LAYER-HANDOVER-SUBSCRIBE(Des-eNodeB-ID, UE Node id) to the xAS. The xAS will then update its list with this information and response the Destination eNodeB with an APPLICATION-LAYER-HANDOVER-SUBSCRIBE-ACK. Afterwards, the Destination eNodeB can create a duplicated node in the overlay by using REQ:MESSAGE-CREATE-DUP-NODE. The duplicated node will manage the data forwarding from the old node while the old node is performing its handover. A response is sent when the duplication is finished and confirmed. A new Resource Reservation process is made for the UE to be attached to the new eNodeB. At this time, two representing nodes are maintained by the two XeNodeBs in the ALM group in order to assure a soft handover process. Even though there are two nodes in the overlay, they are actually sending and receiving identical data. The identical nodes send and receive

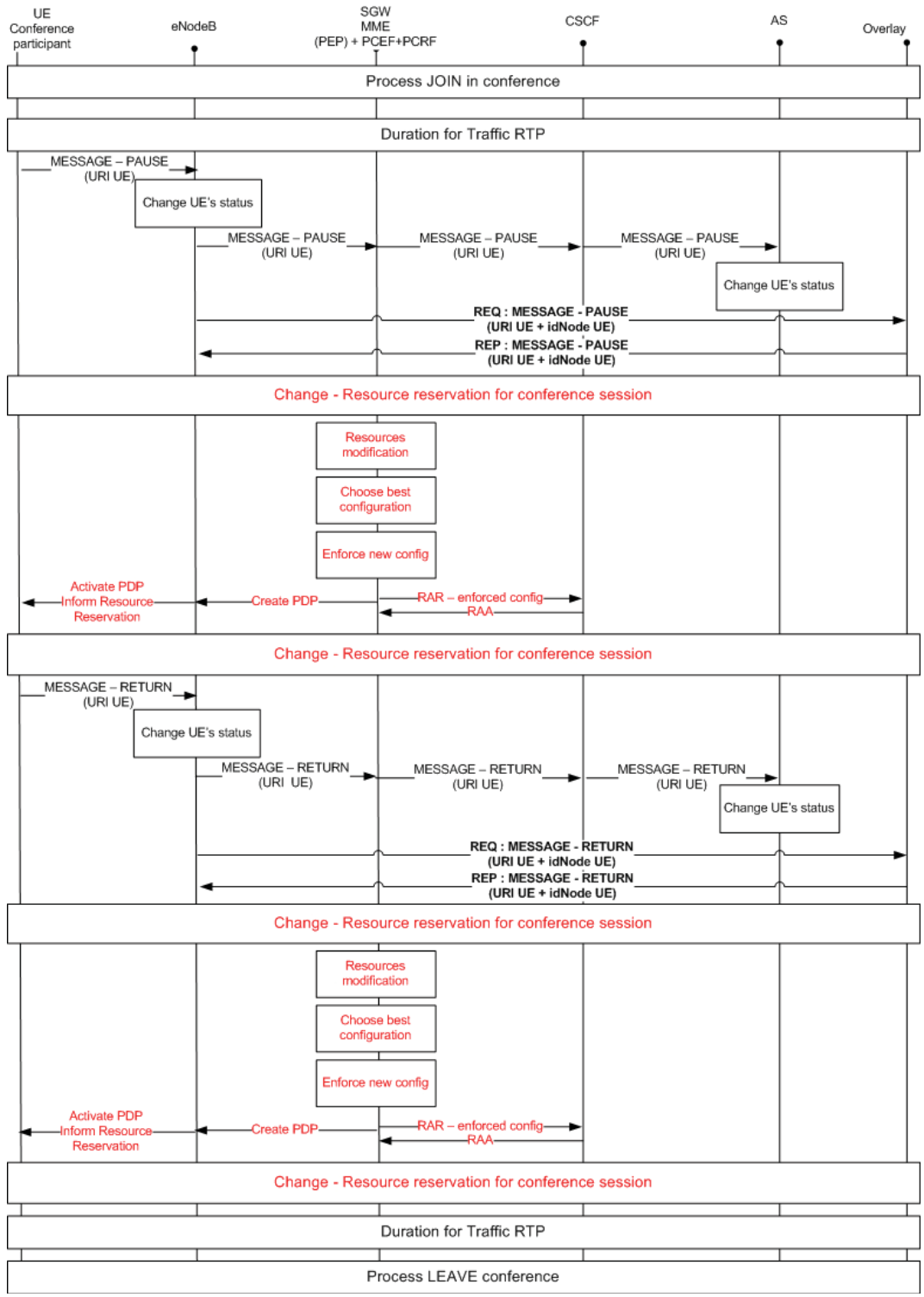


Figure 5.9: The UE Pause/Return process over the distributed video conferencing service with QoS support.

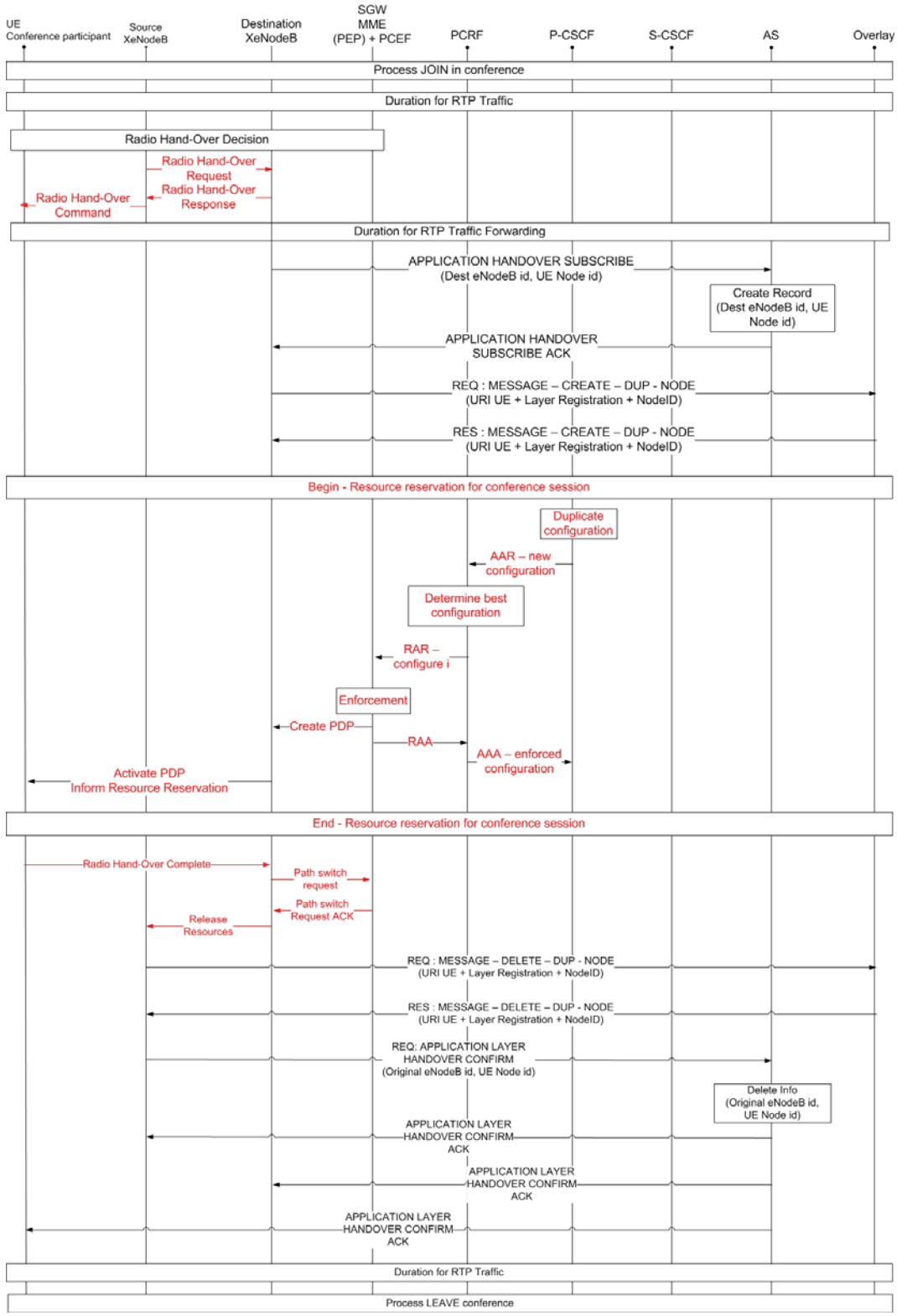


Figure 5.10: Handover operation at the service layer (after the handover at the radio links has been completed) in a distributed conference with QoS support.

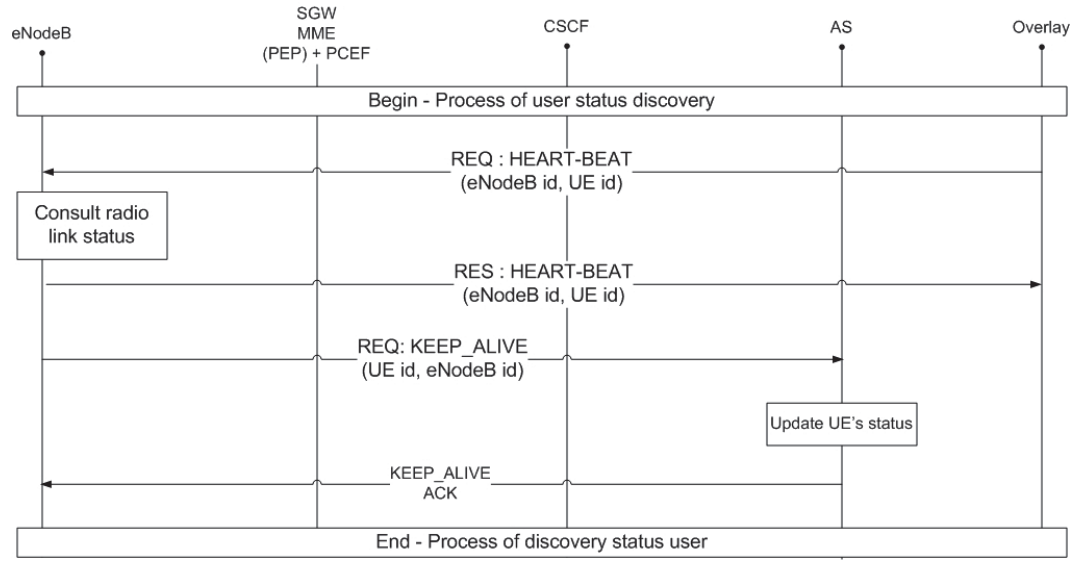


Figure 5.11: Handling heartbeat operation in a distributed conference.

bit-streams as any other normal node in the ALM group thus helping the UE to maintain its conference service via both eNodeB/xBS until the old connection is actually broken. However, only the duplicated node has to manage the data forwarding required by the overlay network. After the handover is finished at the radio layer, the source XeNodeB sends a DELETE-DUP-NODE(UE id, Layer Registration, eNode ID) message to the ALM group and waits to receive the response to confirm that all is finished. It sends a confirmation message to the xAS so that it can be updated. Information regarding the old eNodeB will then be removed from the xAS. The old UE-ID is then removed from the ALM group and only one UE-ID will represent the UE in the conference. The handover process at the service layer is then finished and confirmed. Figure 5.11 shows the heartbeat handling process in the distributed conference with the support of the link measurement process available at the eNodeB. Heartbeat is the mechanism widely applied in an overlay/distributed architecture, in which peers periodically send short messages to inform other peers about their existence in the overlay. It also receives heartbeats from other peers to update about their existence. Many ALM algorithms have to depend on a heartbeat mechanism to maintain their group. If a UE has to directly respond to all heartbeats it receives, it will soon run out of power and computational capacity. Meanwhile, all information regarding a UE's availability is available at the managing eNodeB, a simple link measurement interrogation operation triggered by the xAS can resolve this problem. Therefore, the XeNodeB will periodically consult the link status to query for the availability of all UEs that are joining the distributed conference under its representative and report back to the xAS. The XeNodeB will then send back the heartbeats of all the UEs it is representing to the ALM group to signal their availability. For floor control purposes, the ALM-based distributed conference will

update the list of peers participating in the conference to the xAS.

5.4 Prototype and performance evaluation

5.4.1 Evaluation method

A prototype designed to evaluate the proposal's performance and its feasibility was developed and implemented. The prototype is publicly available at [109]. Figure 5.12 shows the prototype's architecture in which 4 UEs are participating in an ALM-based video conferencing service using a LTE/WiMAX infrastructure with IMS support. We used OpenIMS [110] as the IMS core and the Mobicents platform to build the xAS and XeNodeB. A distributed conferencing service was built based on [45]. A "Rendez-vous" point in the overlay and the XeNodeBs are equipped with a SIP interface so that the overlay can communicate with the xAS via the IMS core. We constructed 3 evaluation scenarios run with different numbers of participants, in which four UEs participate via the IMS core:

- Scenario 1: Centralized IMS-based video conferencing using MRFP, as recommended in the standards (i.e. LTE, WiMAX, IMS);
- Scenario 2: Conventional ALM-based distributed video conferencing service; and
- Scenario 3: Our proposed IMS-based distributed video conferencing service for LTE/WiMAX networks.

Scenario 1 is widely used as the standard architecture to provide the video conferencing services over wireless mobile networks.
Scenario 2 can be referred to as the Web-NGN converged multimedia conferencing system. The most troublesome problem with this scenario (and which is investigated further in the evaluation) is that, the mobile terminals are obligated to handle all the unnecessary signaling and data forwarding traffic that any other peer in the ALM group may handle (even though some peers may be work-stations on fixed networks with unlimited power, high computational capacity and high bandwidth connections). Therefore, the limited power of the mobile terminals will be rapidly used up and their poor radio resources filled up almost exclusively by the ALM's unnecessary signaling and data forwarding traffic.
We selected these three scenarios because they help to explain how our proposal (scenario 3) can sharply reduce the redundant signaling and data forwarding traffic compared to the situation with the conventional overlay-based architecture (scenario 2). By comparing our solution, scenario 3, with the standard centralized solution (scenario 1), we show that our proposed scenario has an equivalent traffic to the centralized architecture but with all the advantages (delay, flexibility, scalability, lower cost, single point of failure) of a distributed architecture as stated in section 5.2.
In all three scenarios, each peer sends 100 audio packets for audio conferencing and 300 video frames for video conferencing. The data was measured three times for each number

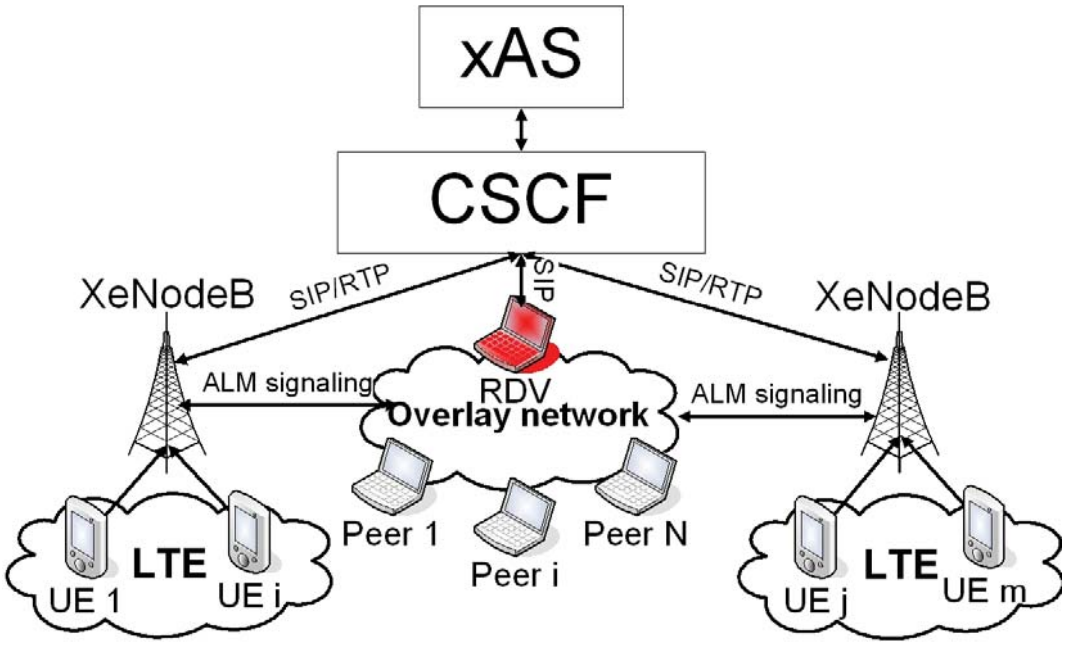


Figure 5.12: Architecture of the prototype.

of participants in each scenario for convergence purposes. In each of these scenarios, we built and applied:

- A prototype of our proposed distributed architecture with the sample nodes in the overlay network;
- A prototype of the proxy nodes running on BSs;
- A real audio conference open source software (sample audio packets obtained from real audio conference sessions),
- A prototype of a video conferencing service on the simple overlay network and our proposed architecture;
- A real SIP client; and
- An open source IMS platform (OpenIMS).

5.4.2 Evaluation results

The numerous advantages of the distributed architecture over the centralized architecture have been elaborated elsewhere, the main purpose of our research is not to compare our solution's performance with that of centralized architecture and prove that our solution

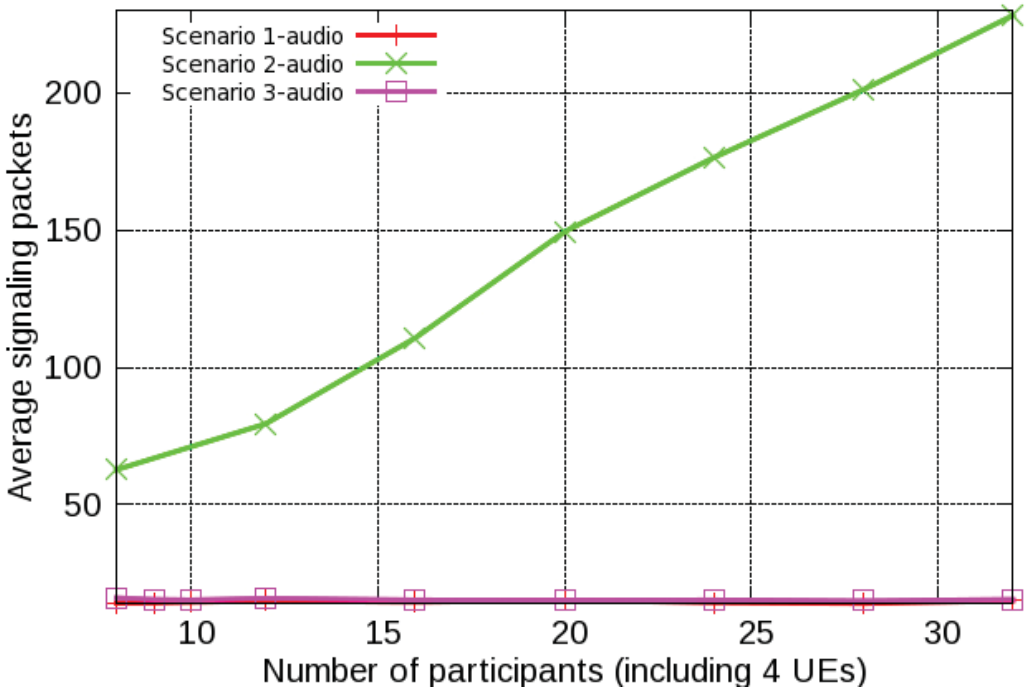


Figure 5.13: Average number of signaling messages required during an audio conferencing session for each of the three scenarios.

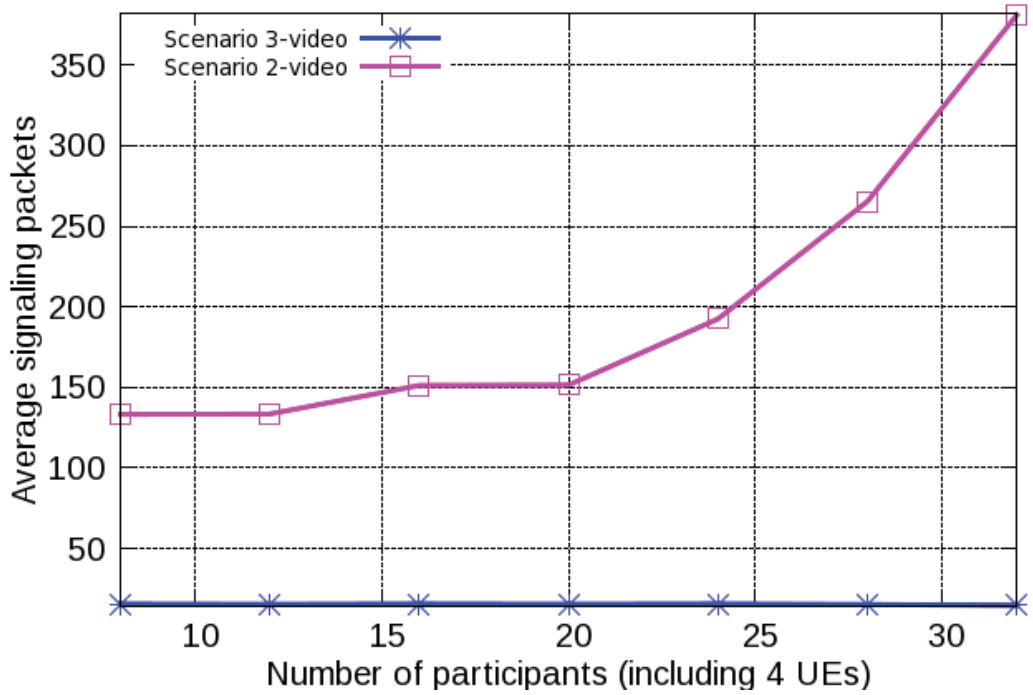


Figure 5.14: Average number of signaling messages required during a video conferencing session for scenarios 2 and 3.

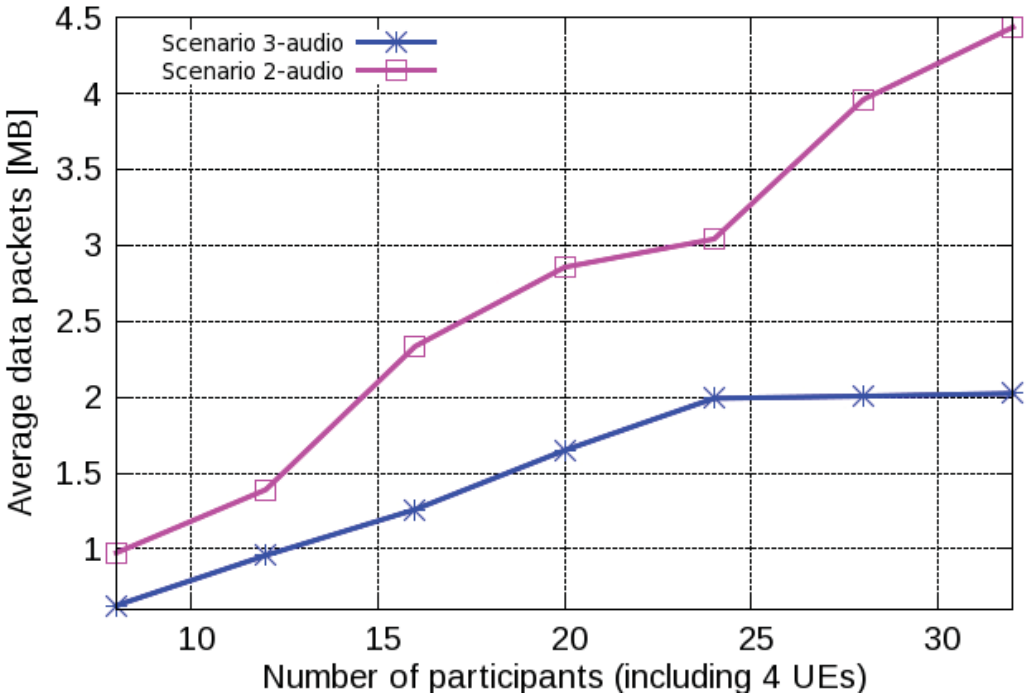


Figure 5.15: Average number of data packets sent/received at each participant during an audio conferencing session for scenarios 2 and 3.

has a better performance. Once again, our purpose is to enable video conferencing service via distributed architecture for next generation wireless networks with only a slight modification of the 4G infrastructure and the IMS platform. To establish the distributed architecture over a mobile network, we have to resolve the crucial problem posed by requiring a high bit rate of wireless terminals. Therefore, we evaluate the performance of our proposal in terms of bit-rates for data and signaling plans.
Figure 5.13 and 5.14 show the average number of signaling messages calculated at UE interfaces during an audio/video conferencing session for all three scenarios (Fig.5.13) and for scenarios 2 and 3 (Fig.5.14). The result shows that scenario 2 needs to use much more signaling packets than scenario 3 to maintain a distributed conference. Our solution, scenario 3 performs better because most of the signaling loads have been processed by the XeNodeBs.
Figure 5.15 and 5.16 show the comparison between the average data traffic monitored on UEs in scenarios 2 and 3 for an audio/video conference. Apparently, data traffic in scenario 3 is lower than in scenario 2 because the XeNodeBs have automatically routed the forwarding traffic for its managing UEs in the overlay. Therefore, the UEs only have to process the data traffic that is intentionally sent to them. We can hereby confirm that our scenario 3 has sharply reduced the traffic that mobile terminals have to contribute to maintain the overlay network.

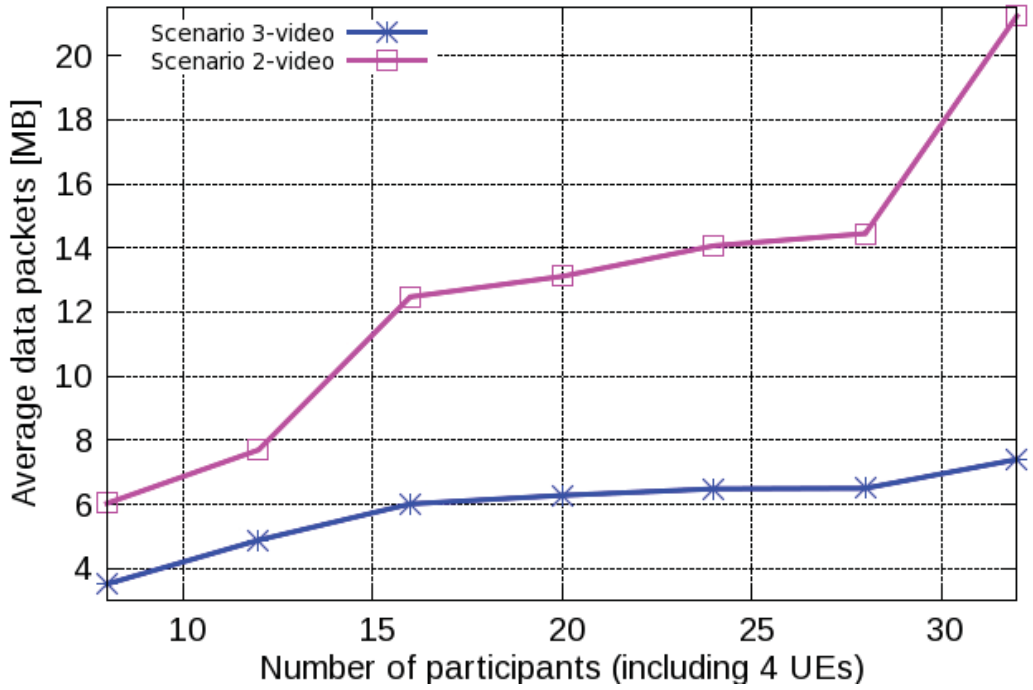


Figure 5.16: Average number of data packets sent/received to/by each participant during a video conferencing session in scenario 2 and scenario 3.

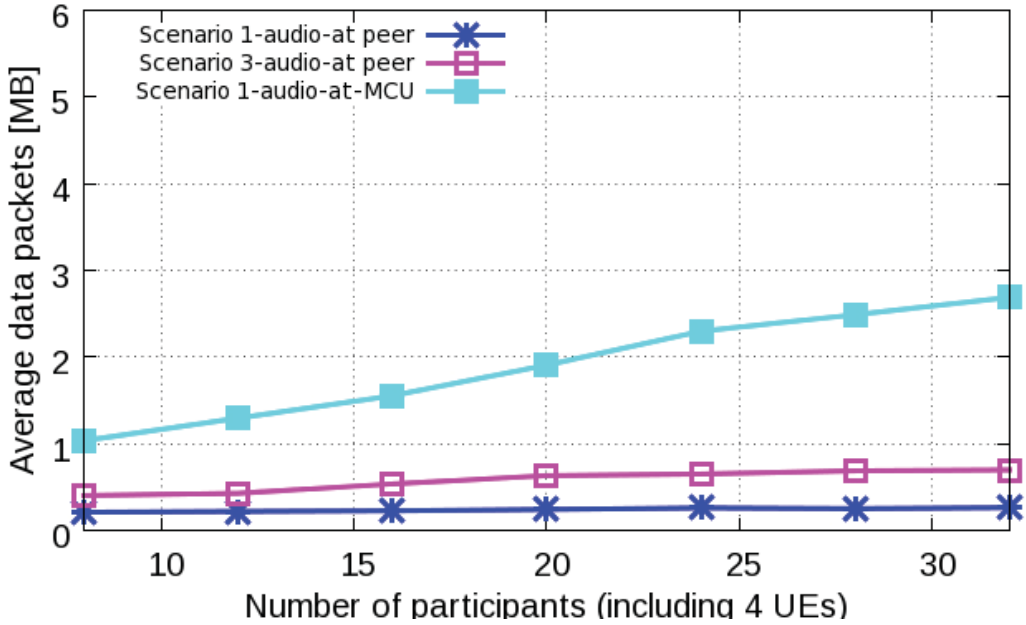


Figure 5.17: Average number of data traffic sent/received to/by each participant/MCU during an audio conferencing session of scenario 1 and scenario 3.

Figure 5.13 also shows that scenarios 1 and 3 require a similar number of signaling packets on UEs. We can conclude that our scenario 3 imposes a requirement similar to that of the standardized centralized scenario, but with all benefits of the distributed architecture that are so obviously lacking in centralized architecture.
Figure 5.17 shows that, the average data traffic at a UE in scenario 3 is slightly higher than the average UE traffic in scenario 1 but far less than the MCU's average traffic in scenario 1. Apparently, the signaling and data traffic in scenario 1 is the smallest because it does not have to maintain a distributed architecture and the MCU is in charge of almost everything which comes with an excessively high cost. Our proposed scenario 3 has fulfilled our design requirements to reduce the data traffic at wireless terminal while maintained all the benefits of a distributed architecture.

5.5 Conclusion

This chapter proposes a new architecture for the inter-connectivity between UEs running on the LTE/WiMAX infrastructure and participating in an overlay-based distributed conference. Experimental results from the prototype have shown a great reduction in signaling traffic as well as in the data traffic handled by each UE and by the core network. The average bit rate required at the wireless terminals in our distributed architecture is equivalent to that of centralized architecture while it is much less than that required for conventional overlay networks.

Table 5.1: Acronyms

3GPP	3rd Generation Partnership Project
4G	4^{th} Generation Network
AAA	Authentication, Authorization, and Accounting
AS	Application Server
ASN	Access Service Network
ASP	Access Service Provider
BS	Base Station
BSC	Base Station Controller
CSN	Connectivity Service Network
CSCF	Call Session Control Function
CAPEX	Capital Expenditures
CDN	Content Delivery Network
eNodeB	Evolved Node B
FCS	Floor Control Server
FCX	Floor Control Mixer
GGSN	Gateway GPRS Support Node
GPRS	General Packet Radio Service
I-CSCF	Interrogating â CSCF
IMS	IP Multimedia Subsystem
MCU	Multipoint Control Unit
MGCF	Media Gateway Controller Function
MGW	Media Gateway
MRF	Media Resource Function
MRFC	Media Resource Function Controller
MRFP	Media Resource Function Processor
MS	Mobile Station
OPEX	Operational Expenditure
P2P	Peer to Peer
P-CSCF	Proxy â CSCF
PCRF	Policy and Charging Rule Function
QoS	Quality of Service
RNC	Radio Network Controller
RTP	Real-time Transfer Protocol
S-CSCF	Serving â CSCF
SN	Super Node
SGSN	Serving GPRS Support Node
VC	Video Conference
xBS	Extended Base Station

Various conferencing scenarios such as join/leave, pause/return, application layer handover, and heartbeat have been considered in the prototype. The main contribution is that it enables an overlay-based video conference with all the benefits of a distributed architecture and without the disadvantages of too much traffic stressing the mobile terminals' wireless connections (an mandatory criterion for wireless networks). The bit-rate required of wireless peers in our proposed distributed architecture is equivalent to that of a standard centralized case (based on results obtained from evaluating our prototype of audio and video distributed conferencing services). The results confirm that it is possible to apply distributed architectures in next generation of wireless networks. Taking advantage of 4G's BSs means that the total cost of the conferencing services can be reduced and that these services can be provided with a minimum modification of relevant standards. The proposal replaces the standard centralized architecture of the IMS-based conference by a more robust solution utilizing intelligence and computational capacity of 4G's BS(s). The prototype of our proposed distributed architecture for a multimedia service has shown that it can be integrated well into a 4G network. Its distributed nature leads to a considerable reduction in cost as well as more flexibility and scalability combined with smaller delay. The architecture can also be applied in WiMAX networks with a reasonable amount of modifications. The security for this distributed architecture can be inherited from the authentication and encryption mechanisms applied by LTE, WiMAX and SIP technologies.

Our solution profits from all the advantages of distributed architecture, such as flexibility, more scalability, smaller delay and lower cost. As a trade-off, some of the computational capacity of the infrastructure (core network and eNodeB/xBS) is required to enable distributed architecture in a mobile network. However, this situation is not really a disadvantage because the counterpart computational capacity required at the centralized media server (MRFC) is now distributed over the network infrastructure (eNodeB(s)/xBS(s) and the core network).

The only drawback of this solution is the computational/intelligence requirements demanded of Base Stations. In 2G/3G, this solution might not be possible because the BSs were just simple antennas with very little capacity and intelligence and the Base Station Controllers (BSC) are separated from BSs. In 4G, the BSC and the BSs are integrated into one entity the xBS (in WiMAX) or the eNodeB (in LTE). Therefore, the intelligence and capacity of the BSC is now available in xBS/eNodeB. They can now support a distributed architecture and make our proposal a realizable goal.

Chapter 6

Conclusion

In conclusion, through the study in this thesis, we have successfully solved the scalability problems which are existing on the multi-party, multimedia conferencing services. As we count on the scalable video coding as the main video codec for our **scalable overlay network**-based distributed video conferencing architecture to solve the **terminal scalability**, we have successfully built an evaluation platform (EvalSVC) for evaluating Scalable Video transmission performance in various network environment such as bottle-neck, overlay, and especially for comparing between centralized an distributed conferencing service architectures. Based on the obtained results from the evaluation process, it is clear that the distributed architecture built on Application Layer Multicast has many advantages over the conventional centralized architecture built on Multipoint Control Unit especially when a multi-layer video coding such as Scalable Video Coding is applied. Obtaining that fact from the evaluation results, the thesis has proposed a new multi-variable cost function based on the requirements of the application. The mathematical derivation process has also been described in details so that one can apply it to obtain other multi-variable cost functions according to their specific requirements. The main idea is to support Application Layer routing algorithms with Application layer's requirements and approach. The newly proposed cost function has considered dynamic requirements from both the application and the underlay network and made a cross-layer optimization for the newly proposed multi-variable cost function. Theoretical analysis has shown that the newly proposed cross-layer multi-variable cost function can simultaneously consider various requirements from the application as well as the possible resource from the network. Intensive evaluation results with real WiMAX access network have shown that, in a real wireless condition, and with ALM-based multicast services, the newly proposed multi-variable cost function can still adapt better to the fast changing available resources as well as the dynamic application's requirements than conventional cost functions. However, while a multi-variable cost function can consider many QoS parameters at the same time, it should be noted that the multi-variable cost function does not always give a better result than the single-variable cost function in some special cases. The newly proposed EvalSVC platform and the multi-variable cost function are a combination of innovative solution for creating and evaluating future multimedia services based on Application Layer Multicast and Scalable Video Coding for new wireless access technologies such as WiMAX.

In order to solve the **architecture scalability**, a new enriched distributed video conferencing architecture considering the limitation of human perception has been proposed.

Analysis mathematical models have been built and compared for centralized, perception-based centralized and perception-based distributed architectures using queuing theory in terms of total waiting time, point-to-point delay, required service rates, and total throughput. It is worth noticing that all the enriched features of the proposed video conferencing architecture have been modelled in details and included in the mathematical analysis and expressions. The numerical simulations obtained from the theoretical analysis models and the off-line statistical data have been done in the context of a multi-party multi-layer video conferencing service. The simulation results show that the newly proposed perception-based distributed architecture can effectively reduce the total waiting time, the end-to-end delay, required service rate in comparison with the centralized and perception-based centralized architectures. Regarding the total throughput, the perception-based distributed architecture have an equivalent traffic with the centralized and perception-based centralized architectures. When the perception-based mechanism is applied, the total throughput of the proposal is even lower than the two conventional architectures. The theoretical analysis result has shown a great advantage of the perception-based distributed architecture against the centralized architecture and the perception-based centralized architecture especially when the total number of conferencing participants increases.

Finally, to solve the **scalable connectivity** between wired line infrastructure (represented by the Internet based participants) and the wireless based mobile infrastructure (represented by the advanced 4G WiMAX/LTE based participants), we proposes a new architecture for the inter-connectivity between UEs running on the LTE/WiMAX infrastructure and participating in an overlay-based distributed conference. Experimental results from the prototype have shown a great reduction in signaling traffic as well as in the data traffic handled by each UE and by the core network. The average bit rate required at the wireless terminals in our distributed architecture is equivalent to that of centralized architecture while it is much less than that required for conventional overlay networks. Various conferencing scenarios such as join/leave, pause/return, application layer handover, and heartbeat have been considered in the prototype. The main contribution is that it enables an overlay-based video conference with all the benefits of a distributed architecture and without the disadvantages of too much traffic stressing the mobile terminals' wireless connections (an mandatory criterion for wireless networks). The bit-rate required of wireless peers in our proposed distributed architecture is equivalent to that of a standard centralized case (based on results obtained from evaluating our prototype of audio and video distributed conferencing services). The results confirm that it is possible to apply distributed architectures in next generation of wireless networks. Taking advantage of 4G's BSs means that the total cost of the conferencing services can be reduced and that these services can be provided with a minimum modification of relevant standards. The proposal replaces the standard centralized architecture of the IMS-based conference by a more robust solution utilizing intelligence and computational capacity of 4G's BS(s). The prototype of our proposed distributed architecture for a multimedia service has shown that it can be integrated well into a

4G network. Its distributed nature leads to a considerable reduction in cost as well as more flexibility and scalability combined with smaller delay. The architecture can also be applied in WiMAX networks with a reasonable amount of modifications. The security for this distributed architecture can be inherited from the authentication and encryption mechanisms applied by LTE, WiMAX and SIP technologies. Our solution profits from all the advantages of distributed architecture, such as flexibility, more scalability, smaller delay and lower cost. As a trade-off, some of the computational capacity of the infrastructure (core network and eNodeB/xBS) is required to enable distributed architecture in a mobile network. However, this situation is not really a disadvantage because the counterpart computational capacity required at the centralized media server (MRFC) is now distributed over the network infrastructure (eNodeB(s)/xBS(s) and the core network). The only drawback of this solution is the computational/intelligence requirements demanded of Base Stations. In 2G/3G, this solution might not be possible because the BSs were just simple antennas with very little capacity and intelligence and the Base Station Controllers (BSC) are separated from BSs. In 4G, the BSC and the BSs are integrated into one entity the xBS (in WiMAX) or the eNodeB (in LTE). Therefore, the intelligence and capacity of the BSC are now available in xBS/eNodeB. They can now support a distributed architecture and make our proposal a realizable goal.
In summary, for multimedia services built over a convergent network with different types of users' terminals, we found that, distributed architecture is better than the centralized architecture specifically in terms of total waiting time, end-to-end delay, required service rates, and total throughput. A cross-layer optimization can offer many advantages because it can model and optimize more accurately both the application's requirements and the network's available resource. This cross-layer optimization can reduce the unnecessary traffic transmitted over the network during a communication session. When applied in a multi-variable cost function, this cross-layer optimization can build an overlay network optimized for multimedia contents delivery. Scalability is an essential feature for the multimedia services to overcome the common congestion problems of the transmission network. We have also proposed a solution for the multimedia services built over a convergent network taken into account all of the distributed architecture, cross-layer optimization and scalability. We have also proved that it is a possible solution which can be integrated into the most up-to-date convergent network infrastructure.

Bibliography

[1] J. R. Wilcox, *Videoconferencing & interactive multimedia: the whole picture*, Cmp, 2000. 3

[2] C. Fu, F. Khendek, and R. Glitho, "Signaling for multimedia conferencing in 4G: the case of integrated 3G/MANETs," *IEEE Communications Magazine*, vol. 44, no. 8, pp. 90–99, 2006. 3

[3] J. V. Team, "Advanced video coding for generic audiovisual services," *ITU-T Rec. H*, vol. 264, pp. 14496–10. 3

[4] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H. 264/AVC standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103–1120, 2007. 4

[5] J. Klaue, B. Rathke, and A. Wolisz, "Evalvid-a framework for video transmission and quality evaluation," *Lecture notes in computer science*, pp. 255–272, 2003. 5

[6] Tien A. Le and Hang Nguyen, "Application-aware cost function and its performance evaluation over scalable video conferencing services on heterogeneous networks," in *2012 IEEE Wireless Communications and Networking Conference: Mobile and Wireless Networks (IEEE WCNC 2012 Track 3 Mobile & Wireless)*, Paris, France, Apr. 2012. 5, 6

[7] Tien A. Le, Hang Nguyen, and Hongguang Zhang, "Scalable Video transmission on overlay networks," in *Second International Conferences on Advances in Multimedia*, Athens, Greece, June 2010, pp. 180–184. 5, 9, 30

[8] Tien A. Le, Hang Nguyen, and Hongguang Zhang, "EvalSVC - An evaluation platform for scalable video coding transmission," in *Consumer Electronics (ISCE), 2010 IEEE 14th International Symposium on*, 2010, pp. 1–6. 6, 44

[9] Tien A. Le, Quang H. Nguyen, and Anh M. Nguyen, *EvalSVC tool-set: http://code.google.com/p/evalsvc/*, 2009. 6, 31

[10] C. Bouras, S. Charalambides, K. Stamos, S. Stroumpis, and G. Zaoudis, "Power Management for SVC Video over Wireless Networks," in *Broadband and Wireless Computing, Communication and Applications (BWCCA), 2011 International Conference on*, Oct. 2011, pp. 270–276. 6

[11] P. McDonagh, C. Vallati, A. Pande, P. Mohapatra, P. Perry, and E. Mingozzi, "Quality-Oriented Scalable Video Delivery using H. 264 SVC on an LTE Network," . 6

[12] K. Rantelobo, G. Hendrantoro, A. Affandi, and H. A. Zhao, "A New Scheme for Evaluating Video Transmission over Broadband Wireless Network," *Future Wireless Networks and Information Systems*, pp. 335–341, 2012. 6

[13] A. Pande, V. Ramamurthi, and P. Mohapatra, "Quality-oriented Video delivery over LTE using Adaptive Modulation and Coding," . 6

[14] R. Skupin, C. Hellge, T. Schierl, and T. Wiegand, "Packet level video quality evaluation of extensive H. 264/AVC and SVC transmission simulation," *Journal of Internet Services and Applications*, pp. 1–10, 2011. 6

[15] C. H. Ke, "myEvalSVC-an Integrated Simulation Framework for Evaluation of H. 264/SVC Transmission," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 6, no. 1, pp. 377–392, 2012. 6

[16] Pejman Goudarzi, "Scalable video transmission over multi-hop wireless networks with enhanced quality of experience using swarm intelligence," *Signal Processing: Image Communication*, , no. 0, 2012. 6

[17] T. A. Le and H. Nguyen, "Centralized and distributed architectures of scalable video conferencing services," in *Ubiquitous and Future Networks (ICUFN), 2010 Second International Conference on*. IEEE, 2010, pp. 394–399. 6

[18] T. A. Le, H. Nguyen, and H. Zhang, "Scalable Video transmission on overlay networks," in *Advances in Multimedia (MMEDIA), 2010 Second International Conferences on*. IEEE, 2010, pp. 180–184. 6

[19] M. Hu, H. Zhang, T. A. Le, and H. Nguyen, "Performance evaluation of video streaming over mobile WiMAX networks," in *GLOBECOM Workshops (GC Wkshps), 2010 IEEE*. IEEE, 2010, pp. 898–902. 6

[20] Quoc T. Tran, Tien A. Le, and Hang Nguyen, "WiMAX-based overlay conferencing service," in *Intelligence in Next Generation Networks (ICIN), 2011 15th International Conference on*, Oct. 2011, pp. 17–22. 6

[21] S. E. Deering and D. R. Cheriton, "Multicast routing in datagram internetworks and extended LANs," *ACM Transactions on Computer Systems (TOCS)*, vol. 8, no. 2, pp. 85–110, 1990. 8, 85

[22] C. Diot, B. N. Levine, B. Lyles, H. Kassem, and D. Balensiefen, "Deployment issues for the IP multicast service and architecture," *IEEE Network*, vol. 14, no. 1, pp. 78–88, 2000. 8, 13, 85

[23] R. Boivie, N. Feldman, Y. Imai, W. Livens, D. Ooms, and O. Paridaens, "Explicit multicast (Xcast) concepts and options," *Request for Comments (RFC)*, vol. 5058, 2007. 8

[24] J. N. Hwang, "Multimedia Networking: From Theory to Practice," 2009. 8

[25] M. Hosseini, D. T. Ahmed, S. Shirmohammadi, and N. D. Georganas, "A survey of application-layer multicast protocols," *IEEE Communications Surveys & Tutorials*, vol. 9, no. 3, pp. 58–74, 2007. 9

[26] K. Tirasoontorn, S. Kamolphiwong, and S. Sae-Wong, "Distributed P2P-SIP conference construction," in *Proceedings of the International Conference on Mobile Technology, Applications, and Systems*. ACM, 2008, p. 20. 12

[27] S. Firestone, T. Ramalingam, and S. Fry, *Voice and video conferencing fundamentals*, Cisco Press, 2007. 12

[28] T. M. O'Neil, "Quality of experience and quality of service for IP video conferencing," *Polycom Video Communications, Milpitas, CA, USA, White paper*, 2002. 12

[29] I. Ekin Akkus, "O "Ozkasap, and M. Reha Civanlar, "Peer-to-peer multipoint video conferencing with layered video," *Journal of Network and Computer Applications*, 2010. 12

[30] Y. Lu, Y. Zhao, F. Kuipers, and P. Van Mieghem, "Measurement study of multiparty video conferencing," *NETWORKING 2010*, pp. 96–108, 2010. 12

[31] Eleftheriadis Alexandros, Civanlar M. Reha, and Shapiro Ofer, "Multipoint videoconferencing with scalable video coding," *Journal of Zhejiang University - Science A*, vol. 7, no. 5, pp. 696–705, 2006. 12

[32] L. De Cicco, S. Mascolo, and V. Palmisano, "Skype video responsiveness to bandwidth variations," in *Proceedings of the 18th International Workshop on Network and Operating Systems Support for Digital Audio and Video*. ACM, 2008, pp. 81–86. 13

[33] S. A. Baset and H. Schulzrinne, "An analysis of the skype peer-to-peer internet telephony protocol," *Arxiv preprint cs/0412017*, 2004. 13

[34] R. Spiers and N. Ventura, "An Evaluation of Architectures for IMS Based Video Conferencing," *University of Cape Town, Rondebosch South Africa*, 2009. 13

[35] M. S. Silver, "Browser-based applications: popular but flawed?," *Information Systems and E-Business Management*, vol. 4, no. 4, pp. 361–393, 2006. 13

[36] S. E. Deering, "Multicast routing in a datagram internetwork," 1991. 13

[37] M. R. Civanlar, Ö, and T. Çelebi, "Peer-to-peer multipoint videoconferencing on the Internet," *Signal Processing: Image Communication*, vol. 20, no. 8, pp. 743–754, 2005. 13

[38] C. Luo, W. Wang, J. Tang, J. Sun, and J. Li, "A Multiparty Videoconferencing System Over an Application-Level Multicast Protocol," *IEEE Transactions on Multimedia*, vol. 9, no. 8, pp. 1621–1632, 2007. 13, 14, 85

[39] I. E. Akkus, M. R. Civanlar, and O. Ozkasap, "Peer-to-Peer Multipoint Video Conferencing using Layered Video," in *Image Processing, 2006 IEEE International Conference on*, Oct. 2006. 14

[40] M. Ponec, S. Sengupta, M. Chen, J. Li, and P. A. Chou, "Multi-rate peer-to-peer video conferencing: A distributed approach using scalable coding," in *Proceedings of the 2009 IEEE international conference on Multimedia and Expo*. IEEE Press, 2009, pp. 1406–1413. 14

[41] Y. Chu, S. Rao, S. Seshan, and H. Zhang, "Enabling conferencing applications on the internet using an overlay muilticast architecture," in *Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications*. ACM New York, NY, USA, 2001, pp. 55–67. 14

[42] J. Lennox and H. Schulzrinne, "A protocol for reliable decentralized conferencing," in *Proceedings of the 13th international workshop on Network and operating systems support for digital audio and video*. ACM, 2003, pp. 72–81. 14

[43] F. Pereira, L. Torres, C. Guillemot, T. Ebrahimi, R. Leonardi, and S. Klomp, "Distributed Video Coding: Selecting the most promising application scenarios," *Signal Processing: Image Communication*, vol. 23, no. 5, pp. 339–352, 2008. 14

[44] H. Jeong, J. Abuan, J. Normile, R. Salsbury, and B. S. Tung, "Heterogeneous video conferencing," May 2011. 14

[45] Tien A. Le and Hang Nguyen, "Centralized and distributed architectures of scalable video conferencing services," in *The Second International Conference on Ubiquitous and Future Networks (ICUFN 2010)*, Jeju Island, Korea, June 2010, pp. 394–399. 14, 16, 93

[46] Tien Anh Le; Hang Nguyen; Hongguang Zhang, "EvalSVC - an evaluation platform for scalable video coding transmission," in *14th International Symposium on Consumer Electronics (ISCE 2010)*, Braunschweig, Germany, June 2010, pp. 85–90. 14, 26

[47] Tien A. Le and Hang Nguyen, "Perception-based Application Layer Multicast Algorithm for scalable video conferencing," in *IEEE GLOBECOM 2011 - Communication Software, Services, and Multimedia Applications Symposium (GC'11 - CSWS)*, Houston, Texas, USA, Dec. 2011. 16, 54

[48] 3rd Generation Partnership Project, "IP Multimedia Subsystem (IMS), Stage 2 (Release 5)," *3GPP TS 23.228*, vol. 5, 2004. 16

[49] S. Wenger, Y. K. Wang, T. Schierl, and A. Eleftheriadis, "RTP payload format for SVC video," *draft, Internet Engineering Task Force (IETF)*, Sept. 2009. 20, 21, 24

[50] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H. 264/AVC video coding standard," *IEEE Transactions on circuits and systems for video technology*, vol. 13, no. 7, pp. 560–576, 2003. 20, 24

[51] S. Wenger, A. G. Teles, and G. Berlin, "H. 264/avc over ip," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 645–656, 2003. 21

[52] S. Wenger, Y. Wang, and M. M. Hannuksela, "RTP payload format for H. 264/SVC scalable video coding," *Journal of Zhejiang University-SCIENCE A*, vol. 7, no. 5, pp. 657–667, 2006. 21

[53] Y. Wang, M. M. Hannuksela, S. Pateux, A. Eleftheriadis, and S. Wenger, "System and transport interface of SVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 149, 2007. 21

[54] J. Reichel, H. Schwarz, and M. Wien, "Joint scalable video model JSVM-8," *ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q. 6, JVT- U*, 2006. 21

[55] J. L. Feuvre, C. Concolato, and J. C. Moissinac, "GPAC: open source multimedia framework," in *MULTIMEDIA'07: Proceedings of the 15th international conference on Multimedia*, 2007. 21

[56] I. Rec, "P. 800: Methods for subjective determination of transmission quality," *International Telecommunication Union*, 1996. 24, 28

[57] C. H. Ke, C. K. Shieh, W. S. Hwang, and A. Ziviani, "An evaluation framework for more realistic simulations of MPEG video transmission," *Journal of Information Science and Engineering*, vol. 24, no. 2, pp. 425–440, 2008. 24

[58] I. Baumgart, B. Heep, and S. Krause, "OverSim: A flexible overlay network simulation framework," in *Proceedings of 10th IEEE Global Internet Symposium (GI'07) in conjunction with IEEE INFOCOM*. Citeseer, 2007, vol. 7, pp. 79–84. 25, 45

[59] M. Reisslein, L. Karam, and P. Seeling, "H. 264/AVC and SVC Video Trace Library: A Quick Reference Guide http://trace. eas. asu. edu," 2009. 25, 29

[60] D. Constantinescu and A. Popescu, "Implementation of Application Layer Multicast in OverSim," in *4th Euro-FGI Workshop on" New Trends in Modelling, Quantitative Methods and Measurements"*. Citeseer. 29, 47

[61] B. M. Waxman, "Routing of multipoint connections," *Selected Areas in Communications, IEEE Journal on*, vol. 6, no. 9, pp. 1617–1622, 2002. 34

[62] B. Fortz and M. Thorup, "Internet traffic engineering by optimizing OSPF weights," in *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*. IEEE, 2002, vol. 2, pp. 519–528. 34

[63] L. H. Sahasrabuddhe and B. Mukherjee, "Multicast routing algorithms and protocols: A tutorial," *Network, IEEE*, vol. 14, no. 1, pp. 90–102, 2002. 34

[64] C. Fang, C. Feng, and X. Chen, "A heuristic algorithm for minimum cost multicast routing in OTN network," in *Wireless and Optical Communications Conference (WOCC), 2010 19th Annual*. IEEE, 2010, pp. 1–5. 34

[65] I. Matta and L. Guo, "On routing real-time multicast connections," in *IEEE International Symposium on Computers and Communications, 1999. Proceedings*, 1999, pp. 65–71. 34, 35

[66] D. H. Lorenz, A. Orda, and D. Raz, "Optimal partition of QoS requirements for many-to-many connections," in *IEEE INFOCOM*. Citeseer, 2003, vol. 3, pp. 1670–1679. 35

[67] D. H. Lorenz and A. Orda, "Optimal partition of QoS requirements on unicast paths and multicast trees," *IEEE/ACM Transactions on Networking (TON)*, vol. 10, no. 1, pp. 102–114, 2002. 35

[68] R. Widyono, "The design and evaluation of routing algorithms for real-time channels," *International Computer Science Institute, TR-94-024*, 1994. 35

[69] A. Bueno, P. Vila, and R. Fabregat, "Multicast extension of unicast charging for qos services," in *Proceedings of 4 th IEEE European Conference on Universal Multiservice Networks (ECUMN)*. Citeseer. 35

[70] Tien A. Le, Hang Nguyen, and Quang H. Nguyen, "Toward building an efficient Application Layer Multicast tree," in *IEEE-RIVF 2010 International Conference on Computing and Telecommunication Technologies*, 2010. 36

[71] K. Calvert and E. Zegura, "GT internetwork topology models (GT-ITM)," 1997. 46

[72] E. W. Zegura, K. L. Calvert, and S. Bhattacharjee, "How to model an internetwork," in *Proceedings IEEE INFOCOM'96. Fifteenth Annual Joint Conference of the IEEE Computer Societies. Networking the Next Generation*, 1996, vol. 2. 46

[73] D. Constantinescu, *Overlay multicast networks: elements, architectures and performance*, Department of Telecommunication Systems, School of Engineering, Blekinge Institute of Technology. 46

[74] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, "Scalable application layer multicast," in *Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications*. ACM, 2002, p. 217. 46

[75] R. Itu-T and I. Recommend, "G. 114," *One-way transmission time*, vol. 18, 2000. 47

[76] Franck Gillet, "Rapport du project POSEIDON," Tech. Rep., June 2011. 48

[77] A. Varga, "INET Framework http://inet.omnetpp.org/," 2007. 48

[78] A. Varga, "OMNeT++ http://www.omnetpp.org/," *IEEE Network Interactive*, vol. 16, no. 4, 2002. 48

[79] Tien A. Le, Hang Nguyen, and Hongguang Zhang, "Multi-variable cost function for Application Layer Multicast routing," in *IEEE Globecom 2010 - Communications Software, Services and Multimedia Applications Symposium (GC10 - CSSMA)*, Miami, Florida, USA, Dec. 2010. 54

[80] K. Boakye, B. Trueba-Hornero, O. Vinyals, and G. Friedland, "Overlapped speech detection for improved speaker diarization in multiparty meetings," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*. IEEE, 2008, pp. 4353–4356. 55

[81] S. Siatras, N. Nikolaidis, M. Krinidis, and I. Pitas, "Visual lip activity detection and speaker detection using mouth region intensities," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 19, no. 1, pp. 133–137, 2009. 55

[82] F. Talantzis, A. Pnevmatikakis, and A. G. Constantinides, "Audio–visual active speaker tracking in cluttered indoors environments," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 38, no. 3, pp. 799–807, 2008. 55

[83] C. Zhang, P. Yin, Y. Rui, R. Cutler, P. Viola, X. Sun, N. Pinto, and Z. Zhang, "Boosting-Based Multimodal Speaker Detection for Distributed Meeting Videos," *Multimedia, IEEE Transactions on*, vol. 10, no. 8, pp. 1541–1552, 2008. 55

[84] D. Gross, *Fundamentals of queueing theory*, Wiley-India, 2008. 59, 61, 62

[85] W. L. Smith, "On the cumulants of renewal processes," *Biometrika*, vol. 46, no. 1/2, pp. 1–29, 1959. 62

[86] W. Whitt, "Approximating a point process by a renewal process, I: Two basic methods," *Operations Research*, pp. 125–147, 1982. 62

[87] S. L. Albin, "Approximating a point process by a renewal process, II: Superposition arrival processes to queues," *Operations Research*, pp. 1133–1162, 1984. 62

[88] W. Whitt, "The queueing network analyzer," *Bell System Technical Journal*, vol. 62, no. 9, pp. 2779–2815, 1983. 62

[89] G. Van der Auwera, P. T. David, and M. Reisslein, "Traffic and quality characterization of single-layer video streams encoded with the H. 264/MPEG-4 advanced video coding standard and scalable video coding extension," *IEEE Transactions on Broadcasting*, vol. 54, no. 3 part 2, pp. 698–718, 2008. 68

[90] Geert Van der Auwera, Prasanth T. David, Martin Reisslein, and Lina J. Karam, "Traffic and quality characterization of the H.264/AVC scalable video coding extension," *Adv. MultiMedia*, vol. 2008, no. 2, pp. 1–27, 2008. 68

[91] B. Y. Choi, S. Moon, Z. L. Zhang, K. Papagiannaki, and C. Diot, "Analysis of point-to-point packet delay in an operational network," *Computer Networks*, vol. 51, no. 13, pp. 3812–3827, 2007. 72

[92] 3gpp, "Conferencing using the IP multimedia (IM)core network (CN) subsystem-ts 24.147," Tech. Rep., Mar. 2008. 84

[93] F. Belqasmi, C. Fu, M. Alrubaye, and R. Glitho, "Design and implementation of advanced multimedia conferencing applications in the 3GPP IP multimedia subsystem," *IEEE Communications Magazine*, vol. 47, no. 11, pp. 156–163, 2009. 84

[94] A. Buono, S. Loreto, L. Miniero, and S. P. Romano, "A distributed IMS enabled conferencing architecture on top of a standard centralized conferencing framework [IP Multimedia Systems (IMS) Infrastructure and Services]," *IEEE Communications Magazine*, vol. 45, no. 3, pp. 152–159, 2007. 84

[95] J. G. Andrews, A. Ghosh, and R. Muhamed, *Fundamentals of WIMAX*, Prentice Hall USA, 2007. 85

[96] Quoc T. Tran, Tien A. Le, and Hang Nguyen, "WiMAX-based overlay conferencing service," in *2011 15th International Conference on Intelligence in Next Generation Networks (ICIN) - From Bits to Data, From Pipes to Clouds (ICIN 2011)*, Berlin, Germany, Oct. 2011. 85

[97] 3gpp, *UTRA-UTRAN Long Term Evolution (LTE) and 3GPP System Architecture Evolution (SAE)*, 2005. 85

[98] P. Beming, L. Frid, G. Hall, P. Malm, T. Noren, M. Olsson, and G. Rune, "Lte-sae architecture and performance," *Ericsson Review*, vol. 3, pp. 98–104, 2007. 85

[99] Tien A. Le, Hang Nguyen, and Noel Crespi, "IMS-based distributed multimedia conferencing service for LTE," in *2012 IEEE Wireless Communications and Networking Conference: Services, Applications, and Business (IEEE WCNC 2012 Track 4 SAB)*, Paris, France, Apr. 2012. 85, 89

[100] Wimax Forum, "WiMAX Forum Network Architecture - Architecture, detailed Protocols and Procedures - Policy and Charging Control," Tech. Rep., Dec. 2009. 85, 86

[101] F. Xu, L. Zhang, and Z. Zhou, "Interworking of Wimax and 3GPP networks based on IMS [IP Multimedia Systems (IMS) Infrastructure and Services]," *Communications Magazine, IEEE*, vol. 45, no. 3, pp. 144–150, 2007. 86

[102] J. Song, S. Lee, and D. Cho, "Hybrid coupling scheme for UMTS and wireless LAN interworking," in *Vehicular Technology Conference, 2003. VTC 2003-Fall. 2003 IEEE 58th.* IEEE, 2003, vol. 4, pp. 2247–2251. 86

[103] X. Wu and Others, "Use of session initiation protocol (SIP) and simple object access protocol (SOAP) for conference floor control protocol (SOAP) for conference floor control," Tech. Rep., Internet draft, Internet Engineering Task Force, 2003. 87

[104] K. Daoud, P. Herbelin, and N. Crespi, "UFA: Ultra Flat Architecture for high bitrate services in mobile networks," in *IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008*, 2008, pp. 1–6. 89

[105] K. Daoud, P. Herbelin, and N. Crespi, "One-Node-Based Mobile Architecture for a Better QoS Control," *IFIP Wireless days*, 2008. 89, 91

[106] W. Zhuang, Y. S. Gan, K. J. Loh, and K. C. Chua, "Policy-based QoS architecture in the IP multimedia subsystem of UMTS," *Network, IEEE*, vol. 17, no. 3, pp. 51–57, 2003. 91

[107] A. Racz, A. Temesvary, and N. Reider, "Handover Performance in 3GPP Long Term Evolution (LTE) Systems," in *Mobile and Wireless Communications Summit, 2007. 16th IST*, 2007. 92

[108] R. Y. Kim, I. Jung, X. Yang, and C. C. Chou, "Advanced handover schemes in IMT-advanced systems [WiMAX/LTE Update]," *Communications Magazine, IEEE*, vol. 48, no. 8, pp. 78–85, 2010. 92

[109] Tien A. Le, Quoc T. Tran, and Quang H. Nguyen, *Multi-P2P prototype: http://code.google.com/p/multi-p2p/*, 2009. 93

[110] F. Fokus, "Open IMS Core," *URL http://www. openimscore. org.* 93