Optimisation des fonctions push-to-talk dans un environnement hybride intégrant LTE et satellite

Joan Ventura Jaume

▶ **To cite this version:**

Joan Ventura Jaume. Optimisation des fonctions push-to-talk dans un environnement hybride intégrant LTE et satellite. Networking and Internet Architecture [cs.NI]. Télécom Bretagne; Université de Bretagne Occidentale, 2016. English. NNT : . tel-01565063

HAL Id: tel-01565063
https://hal.science/tel-01565063

Submitted on 19 Jul 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITE
BRETAGNE
LOIRE

THÈSE / Télécom Bretagne
sous le sceau de l'Université Bretagne Loire

pour obtenir le grade de Docteur de Télécom Bretagne

En accréditation conjointe avec l'Ecole Doctorale Sicma
Mention : Sciences et Technologies de l'Information et de la Communication

présentée par
Joan Ventura Jaume

préparée dans le département Micro-ondes
Laboratoire Labsticc

Optimisation des fonctions push-to-talk dans un environnement hybride intégrant LTE et satellite

Thèse soutenue le 20 Mai 2016
Devant le jury composé de :

Thierry Gayraud
Professeur, UPSITECH – Université Toulouse III / président

André-Luc Beylot
Professeur, ENSEEIHT – IRIT - Toulouse / rapporteur

Javier Francisco González Castaño
Professeur, Université de Vigo - Espagne / rapporteur

Franck Cibaud
Ingénieur, Airbus Defence & Space – Deuil-La-Barre / examinateur

Laurent Franck
Professeur, Télécom Bretagne / directeur de thèse

n° d'ordre : 2016telb0398

THESE / Télécom Bretagne

Sous le sceau de l'Université Bretagne Loire
pour obtenir le grade de

Docteur de Télécom Bretagne

En accréditation conjointe avec l'Ecole Doctorale Sicma

Mention: **Sciences et Technologies de l'Information et de la Communication**

OPTIMISATION DES FONCTIONS PUSH-TO-TALK DANS UN ENVIRONNEMENT HYBRIDE INTÉGRANT LTE ET SATELLITE

Présentée par

Joan Ventura Jaume

Préparée dans le département Micro-ondes

Laboratoire : Télécom Bretagne - Toulouse

Thèse soutenue le 20 Mai 2016

Devant le jury composé de **:**

Thierry Gayraud
Professeur, Université Paul Sabatier - Toulouse III / président
André-Luc Beylot
Professeur, ENSEEIHT / rapporteur
Francisco Javier González Castaño
Professeur, Université de Vigo / rapporteur
Franck Cibaud
Ingénieur, Airbus Defence & Space / examinateur
Laurent Franck
Professeur, Télécom Bretagne / directeur de thèse

THESIS / Télécom Bretagne

Under the seal of the University Bretagne Loire
to obtain the grade of

Doctor of Télécom Bretagne

Accredited jointly with the Sicma Doctoral School

Mention: **Communication Sciences and Information Technology**

OPTIMIZATION OF PUSH-TO-TALK FUNCTIONS IN AN HYBRID ENVIRONMENT INTEGRATING LTE AND SATELLITE

Presented by

Joan Ventura Jaume

Prepared in the Microwaves department

Laboratory: Télécom Bretagne - Toulouse

Thesis presented on the 20th May 2016

In front of the jury composed by**:**

Thierry Gayraud
Professor, University Paul Sabatier - Toulouse III / president
André-Luc Beylot
Professor, ENSEEIHT / reviewer
Francisco Javier González Castaño
Professor, University of Vigo / reviewer
Franck Cibaud
Engineer, Airbus Defence & Space / examiner
Laurent Franck
Professor, Télécom Bretagne / thesis director

Acknowledgements

The present document is the result of the research performed from November 2012 to January 2016 during my PhD thesis in Satellite Telecommunications at Télécom Bretagne and Airbus Defence and Space, formerly the satellite services division of EADS Astrium. This work is the outcome of a successful collaboration between these two entities, to which I am extremely thankful. This has been possible thanks to the funding of the R&D of Airbus DS as well as the grant from the CIFRE (Conventions Industrielles de Formation par la REcherche) program, which is managed by the ANRT (Association Nationale de Recherche Technologie).

My greatest gratitude goes to Prof. Laurent Franck, my academic advisor and thesis director, who gave me an immeasurable support throughout this thesis and provided always the additional details that make this contribution stand out. I had not in mind starting a PhD at the end of my studies, but his first call was the beginning of an incredible opportunity I could not have missed. His managerial, technical opinions and moral support were decisive in the realization of this work.

The people from Airbus Defence and Space deserve the same level of recognition. I sincerely appreciate the advice and help that my two professional supervisors, Franck Cibaud and Laurent Girardeau, have provided me during these years. I thankful for the autonomy they granted me as well as for the directions they gave me to well develop my ideas towards positive results. I am pleased to have been part of the teams that have developed the mobile network activity for the services division of Airbus DS.

I would like to continue by thanking the rest of my colleagues at Airbus. First, I want to express my gratitude to Hervé Fritsch, Renaud Vogeleisen and Florence Dufrasnes for their managerial support, I foresee a very interesting activity in their departments in the following years. Then, I am deeply indebted to Michel Hammann for his aid with my inventions protection. I did not imagine to submit a patent at the beginning of my career. In addition, I would like to show my appreciation to Franck Scholler for supporting the development of my protocol. I am convinced that we will have soon very good news on this subject. Last, but not least, I want to thank Alice de Casanove for making this thesis possible with her link and help with Télécom Bretagne.

I happy to add to this list the friends I have met here in Toulouse. Starting from my flatmates at Bonrepos, then my colleagues from the SCS master, continuing with my catalan friends at Airbus, the runners at Supaero; and finalizing with the people that made me re-experience an erasmus. It is unfortunate that many of you left the city looking for new challenges; you will always have a home close to the Garonne.

Finally, I would like to dedicate the last words of this page to my family. This endeavor to complete a doctorate degree could have never been accomplished without all the love and support from Andrea and my parents over the years. They always helped me see the positive side of each period of this long and not always easy process. I admire their patience and confidence in me.

Abstract

Push-To-Talk (PTT) is a simple communication service that allows a group of people to exchange voice messages. PTT is very popular among law enforcement agents and other emergency services. Its half-duplex nature and regulated operation helps users speak without interruption in an ordered fashion.

Professional Mobile Radio (PMR) systems provide voice and data services to first responders and maintain PTT as their most used feature. Next generation PMR solutions are likely to converge with public land mobile technologies such as Long Term Evolution (LTE) to benefit from higher data rates and better interoperation with other existing networks. During catastrophic events, the supporting infrastructure networks could be destroyed, saturated or absent. Therefore, it is helpful to deploy temporary networks on the isolated areas and backhaul them by means of geostationary satellites to connect to the core backbone. In this thesis, we study the adaptation of the PTT application to this hybrid environment based on satellite links and LTE networks. Legacy solutions were designed without considering the satellite as a key part of the system. Hence, we take the opportunity of the PMR systems evolution to incorporate the challenges of the satellite framework.

The first part of this work addresses the management of PTT group conversations from the control plane standpoint. In PMR, floor control regulates the access to shared channels to ensure only one participant is allowed to speak. In the presence of a high delay satellite links and remotely distributed participants, current mechanisms are challenged and fail to take into account the time ordering of the conversations without compromising the overall delay. Centralized solutions are no longer valid, the presence of satellite links implies a very large disparity of latency between the central server and the users in different locations. We present a distributed floor control protocol tailored to these situations. During the floor determination process, we define *failure* as giving the floor to a user, while he is not eligible as first user to speak. We derive an analytical model that predicts the probability of failure depending on floor control parameters and the system characteristics. Our simulations provide insight about how floor control should be parametrized so to limit failure probability under a certain threshold.

The second part of this document focuses on the optimization of the voice transmission. The establishment of the packet-based networks was a major breakthrough for voice communication services, allowing multiple users to share a common channel without diminishing their quality of experience. However, packetized voice suffers from excessive overhead. As voice packets are usually small compared to other types of information, the portion of the packet that contains the header information can be larger that the one that comprises the voice. Two main techniques have been using in order to reduce the impact of this overhead: frame multiplexing and header compression. Multiplexing gathers multiple voice frames in a single packet, reducing the weight of the header in the total packet size. On the other hand, header compression exploits the redundancy of some of the header fields and reduces the total information sent by consecutive packets. Our approach is to use both techniques in order to reduce the number of bits needed to transmit one voice-based session.

From the user data perspective, VoIP calls and PTT are essentially the same. They share the same application data format and headers in the upper layers. Both services require low jitter and moderate frame error rate. However, PTT is half-duplex and can tolerate a higher latency as long as the message is not interrupted. We propose a scheme that combines multiplexing, header compression and service differentiation between VoIP calls and PTT in order to efficiently use a satellite link while considering the differences in the Quality of Experience (QoE) requirements. Our results show that is a promising manner to double the number of voice sessions managed by the system.

Finally, we examine the resource allocation mechanisms of the satellite system return link and identify the best candidate to regulate the resource demand of terminals managing multiple PTT sessions. We observe that the arrival of multiple calls in a short period of time challenges the allocation procedures and might incur a burst of error. Therefore, we identify the calls that may endanger the beginning of some messages and propose two preventive methods to alleviate the phenomenon.

RESTRICTED DOCUMENT

Abstract

Push-To-Talk (PTT) est un service de communication simple qui permet à un groupe de personnes d'échanger des messages vocaux. PTT est très populaire parmi les forces de l'ordre et d'autres services d'urgence. Sa nature unidirectionnelle et son fonctionnement réglementé permet aux utilisateurs de parler sans interruption de façon ordonné.

Les systèmes de radiocommunications mobiles professionnelles (PMR) fournissent des services voix et donnés aux intervenants des premier secours et maintiennent le PTT comme leur fonction la plus utilisée. Les solutions PMR de la prochaine génération sont susceptibles de converger avec les technologies mobiles terrestres publiques telles que Long Term Evolution (LTE) afin de bénéficier de débits plus élevés et d'une meilleure interopérabilité avec d'autres réseaux existants. Lors d'événements catastrophiques, les infrastructures de télécommunications pourraient être détruites, saturées ou absentes. Par conséquent, il est utile de déployer des réseaux temporaires sur les régions isolées et de les connecter au réseau de coeur à travers de satellites géostationnaires. Dans cette thèse, nous étudions l'adaptation de l'application PTT à cet environnement hybride basé sur des liaisons par satellite et les réseaux LTE. Les solutions existantes ont été conçues sans tenir compte du satellite comme un élément clé du système. Par conséquent, nous saisissons l'occasion de l'évolution des systèmes PMR pour intégrer les défis du cadre des liens satellitaires.

La première partie de cette contribution adresse la gestion des conversations de groupe du type PTT du point de vue du plan de contrôle. Dans les PMR, le *floor control* réglemente l'accès aux canaux partagés pour assurer qu'un seul participant est autorisé à parler. En présence du délai élevé des liaisons par satellite avec les participants répartis à distance, les mécanismes actuels sont contestés et ne parviennent pas à tenir compte de l'ordre temporel des conversations sans compromettre le délai global. Les solutions centralisées ne sont plus valables, la présence d'une liaison par satellite implique une très grande disparité des temps de latence entre le serveur central et les utilisateurs situés dans des endroits différents. Nous présentons un protocole de *floor control* distribué adapté à ces situations. Pendant le processus de détermination de l'utilisateur qui devrait obtenir l'accès, nous définissons *l'échec* comme donner la parole à un utilisateur, alors qu'il n'est pas admissible en tant que premier utilisateur à parler. Nous développons un modèle analytique qui prédit la probabilité d'échec en fonction des paramètres du *floor control* et des caractéristiques du système. Nos simulations donnent un aperçu de la façon dont le protocole doit être paramétré de manière à limiter la probabilité d'échec sous un certain seuil.

La deuxième partie de ce document se focalise sur l'optimisation de la transmission de la voix. La mise en place des réseaux par paquets est une avancée majeure pour les services de communication vocale, permettant à plusieurs utilisateurs de partager un canal commun sans pour autant diminuer la qualité de leur expérience. Cependant, la voix en paquets souffre d'une présence excessive des en-têtes. Comme les paquets de voix sont généralement courts par rapport à d'autres types d'informations, la partie du paquet qui contient les informations d'en-tête peut être plus grande que celle qui comprend la voix.

Deux techniques principales ont été utilisées afin de réduire l'impact de cette surcharge: le multiplexage des trames et de la compression d'en-tête. Le multiplexage rassemble plusieurs trames vocales dans un seul paquet, ce qui réduit le poids de l'en-tête dans la taille totale du paquet. D'autre part, la compression d'en-tête exploite la redondance de certains des champs et réduit la totalité des informations envoyées par des paquets consécutifs. Notre approche consiste à utiliser les deux techniques afin de réduire le nombre de bits nécessaires pour transmettre une session basée sur la voix.

Du point de vue des données d'utilisateur, les appels VoIP et le PTT sont essentiellement le même. Ils partagent le même format des données et des en-têtes dans les couches supérieures. Les deux services nécessitent une faible jitter et un taux d'erreur modéré. Toutefois, le PTT est unidirectionnel et peut tolérer un temps de latence supérieur à condition que le message ne soit pas interrompu. Nous proposons un schéma qui combine le multiplexage, la compression d'en-tête et la différenciation de service entre les appels VoIP et le PTT afin d'utiliser efficacement un lien par satellite tout en tenant compte les différences d'exigences sur la qualité de l'expérience (QoE). Nos résultats montrent qu'il s'agit d'une manière prometteuse de doubler le nombre de sessions vocales gérables par le système.

Finalement, nous examinons les mécanismes d'allocation des ressources de la liaison de retour du système satellite et nous identifions le meilleur candidat pour réguler la demande de ressources pour les terminaux qui gèrent plusieurs sessions PTT. Nous observons que l'arrivée de plusieurs appels dans un court période de temps remet en question les procédures d'attribution et pourrait générer une succession de pertes de paquets. Par conséquent, nous identifions les appels qui peuvent mettre en danger le début de certains messages et nous proposons deux méthodes préventives pour atténuer le phénomène.

RESTRICTED DOCUMENT

Résumé Long

Introduction

Le Push-To-Talk (PTT) est un service de communication simple qui permet à un groupe de personnes d'échanger des messages vocaux. Le concept est explicite: l'utilisateur appuie sur un bouton pour demander la permission de parler et si elle est accordée, il parle et le service délivre le message au reste des participants. La communication est semi-duplex, à sens unique, à l'opposé des services full-duplex tels que les appels vocaux réguliers. En s'assurant qu'une seule personne parle à la fois et en utilisant des mots clés, les utilisateurs peuvent communiquer efficacement. Le PTT est encore le principal moyen de communication entre les agents des forces de l'ordre et d'autres services d'urgence. Dans les dernières décennies, les organismes de sécurité publique et de premiers secours ont bénéficié de réseaux dédiés qui proposent des services tels que PTT. Ces solutions sont communément appelées Professional Mobile Radio (PMR) et se différencient des réseaux mobiles publics en termes d'exigences sur la fiabilité, la disponibilité, la robustesse... L'évolution de la technologie et les coûts associés à des réseaux dédiés ont fait augmenter l'intérêt pour la recherche de nouvelles solutions qui proposent de les fusionner avec les réseaux commerciaux. Avec le déploiement du Long Term Evolution (LTE), les réseaux mobiles ont adopté une architecture IP complète. Des fonctions telles que celles offertes par les PMR et ses exigences sont désormais introduites dans les nouvelles versions de la norme. Les réseaux mobiles et Internet fournissent l'infrastructure nécessaire pour permettre à des millions de personnes de communiquer autour du globe. Cependant, comme toute autre infrastructure sur terre, ils sont exposés à des événements imprévus qui peuvent les endommager et les rendre indisponibles. Les tremblements de terre, les inondations, les pannes de courant, le sabotage humain ou tout simplement la saturation pourraient désactiver ces réseaux et mettre en danger les communications. Dans des situations menaçant la vie, les services de communication sont indispensables et des solutions temporaires doivent être déployées. Les services par satellite peuvent rapidement fournir une couverture mondiale à tout moment et permettre une réponse rapide. Airbus Defence and Space commercialise une solution qui permet le déploiement d'un réseau LTE et l'utilisation d'une liaison par satellite pour communiquer avec le reste du réseau vers l'Internet. Étant donné que la sécurité publique et les agences de premiers secours sont les principaux clients de cette solution, il est devenu nécessaire d'incorporer un service PTT. Notre objectif est d'optimiser les mécanismes conçus pour ces services réglementés dans le monde des réseaux terrestres et de les adapter aux caractéristiques du segment satellite et son interconnexion avec le reste du réseau. L'interconnexion

par satellite suppose un défi pour les systèmes actuels PTT en raison du délai de propagation inhérent et le manque de ressources. La communication entre les utilisateurs des deux côtés de la liaison par satellite pose d'abord un problème pour la gestion du groupe. Il est nécessaire de modifier les protocoles existants afin d'examiner efficacement les différentes latences d'accès. En outre, les ressources sur le lien de retour des systèmes satellitaires doivent être utilisées judicieusement. L'utilisation des mécanismes de demande de capacité (DAMA) améliore l'efficacité en même temps que limite la capacité des applications en temps réel. Il est prévu que d'autres services, tels que le streaming vidéo pour la surveillance, augmentent en importance dans les années suivantes. Par conséquent, il devient crucial de réduire la largeur de bande requise pour les services de transmission de la voix. Le but de ce travail est de concevoir une solution PTT qui se fonde sur deux éléments principaux: un plan de contrôle qui tient compte du retard de la liaison par satellite et une transmission optimale de multiples flux de voix dans le plan des données des utilisateurs.

L'architecture d'une solution PTT sur un réseau hybride LTE et satellite

Les réseaux hybrides sont basés sur deux ou plusieurs technologies de télécommunications. Les différentes options dépendent de l'architecture du réseau, la position de la liaison par satellite et les possibilités de déploiement. Nous proposons une architecture basée sur un réseau hybride temporaire avec backhaul ou interconnexion directe par satellite et une organisation basée sur des serveurs de semi-distribués. Sur l'architecture du réseau, deux possibilités sont proposées. Dans la première nous déployons une radio LTE temporelle qui, a travers du lien satellite, se connecte au réseau de cœur LTE centralisé. Dans la deuxième option, un réseau LTE complet peut être déployé ce qui lui permet d'opérer de façon indépendante. Il est prévu que ces deux options examinées convergent au fil des ans. Les stations radio seront plus capables de fonctionner et de prendre leurs propres décisions sans l'interaction avec un cœur LTE; et la simplification et miniaturisation des entités aidera à développer des cœurs moins chers et plus petits. La gestion des services est répartie entre plusieurs entités, appelées serveurs ou super-nœuds distribués, qui peuvent être situés plus près de l'utilisateur final alors que ceux-ci restent simples. Cela permet une meilleure évolutivité et n'a pas le problème de l'architecture centralisé. Cette solution permet une hiérarchie équilibrée et permet la possibilité d'effectuer des reconfigurations en sous-groupes et de gérer seulement les utilisateurs déployés. Elle introduit également une certaine asymétrie dans le réseau permettant à chaque côté de procéder de façon indépendante et synchroniser lorsque cela est nécessaire. Les super nœuds (SN) ne seront pas seulement responsables de la gestion de la session et du protocole de floor control, mais ils exécuteront les fonctions nécessaires pour optimiser le réseau. Cela est essentiel pour le segment satellite où les super nœuds peuvent exécuter des fonctions similaires à celles effectuées par les proxies en améliorant le rendement. Par exemple, dans le cas de la liaison backhaul par satellite, le super nœud effectue la traduction des flux LTE vers IP. En outre, comme nous le verrons dans cette thèse, ils peuvent être impliqués dans l'optimisation du plan d'utilisateur en ce qui concerne la transmission des paquets vocaux. L'application de terminal d'utilisateur est plus simple dans cette architecture

RESTRICTED DOCUMENT

où tout le traitement complexe et les décisions sont pris au niveau des super nœuds. En outre, un nouvel utilisateur peut facilement accéder au service par simple téléchargement d'une application à partir de son super nœud le plus proche.

Un protocole de floor control distribué pour les PMR de prochaine génération sur des réseaux hybrides LTE et satellite

Dans les Professional Mobile Radio, le floor control réglemente l'accès aux canaux partagés pour assurer qu'un seul participant est autorisé à parler. En présence d'un délai élevé comme celui des liaisons par satellite et avec des participants distribués des deux côtés, les mécanismes actuels sont contestés et ne parviennent pas à prendre en compte l'ordre de la conversation sans compromettre le délai global. L'objectif de cette contribution est d'offrir une solution juste qui respecte la causalité temporelle pour tous les participants, indépendamment de leur emplacement. Les principaux objectifs de notre solution sont la réduction du temps d'accès pour les utilisateurs et assurer que les auditeurs reçoivent les messages presque en même temps. Nous définissons notre protocole comme optimiste, car il donne accès à un l'utilisateur demandant avant de s'assurer qu'il sera le client finalement autorisé. Pendant le processus de détermination du floor, nous définissons échec comme donner la parole à un utilisateur, alors qu'il n'a pas droit d'être le premier utilisateur à parler. Nous dérivons un modèle analytique qui prédit la probabilité d'échec en fonction de des paramètres du floor control et les caractéristiques du système. Nos simulations donnent un aperçu sur comment le floor control doit être paramétré de manière à limiter la probabilité d'échec sous un certain seuil. Chaque requête comprend une timestamp relatif, qui définit la période de temps entre la fin du dernier message et le moment où la nouvelle demande est envoyée. Une demande sera finalement acceptée seulement si son timestamp est le plus petit parmi toutes les demandes reçues pendant la même itération de la phase de contention. La période de contention est le moment où les utilisateurs tentent d'accéder au floor. Elle commence lorsque le premier utilisateur demande les ressources et il se termine lorsque ses messages sont transmis aux auditeurs. Les utilisateurs qui souhaitent parler demandent la parole à leurs super-nœuds (SN). Lorsqu'un SN reçoit une demande locale, il envoie un message pré-acceptation et transmet la demande au reste de SNs. A ce moment, le SN calcule la valeur d'un période de bufferisation ou de mémoire tampon. Dès cet instant, l'utilisateur peut parler et ses paquets seront stockés et transmis aux SN à distance. Ces SN également définissent une période de mémoire tampon quand ils reçoivent la demande. Il est possible que pendant la période tampon une autre demande soit reçue. Dans ce cas, le SN compare les deux timestamp et si celui de la nouvelle demande a une valeur plus petite, cet utilisateur devient le nouveau détenteur temporaire du floor. L'utilisateur parlant est informé par son SN locale et ses messages ne seront pas envoyés au reste des utilisateurs. La durée de la mémoire tampon est ensuite recalculée en fonction des paramètres de la nouvelle demande. Une fois que les périodes expirent, les SN commencent vider la mémoire tampon en transmettant les paquets de voix à leurs utilisateurs à l'écoute. Le SN continue d'enregistrer sur la mémoire tampon les messages de l'émetteur jusqu'à ce que le message de libération ou relâche soit reçu.

RESTRICTED DOCUMENT

Les auditeurs reçoivent le message entier à la même vitesse à laquelle il a été envoyé avec un décalage de temps donné par la période tampon. Un des objectifs de la mémoire tampon est que les différentes SN observent une période de contention similaire. Le temps de bufferisation sera plus court aux SNs situés plus loin des utilisateurs demandant, où le retard expérimenté est supérieur. D'une part, les avantages d'un tel système est que l'utilisateur demandeur peut commencer à parler rapidement une fois confirmé par son SN local; les auditeurs ne voient pas les demandes qui produisent des éventuelles collisions; le temps de bufferisation empêche également l'envoi des messages aux auditeurs avant de connaître le client finalement autorisé; et, enfin, le mécanisme de mémoire tampon permet de transmettre les paquets vocaux rapidement. Grâce au fait que la période de mémoire tampon est adaptée en fonction du retard, tous les auditeurs connaîtront un retard de bouche-à-oreille similaire. D'autre part, les inconvénients sont qu'un utilisateur peut être coupé pour donner accès à un autre; que les périodes de mémoire tampon doivent être optimisées afin d'éviter trop d'erreurs et que l'utilisation des ressources augmente au cours de la période de contention parce que tous les utilisateurs demandant la parole vont envoyer des paquets de voix avant de s'assurer qu'ils sont les utilisateurs autorisés. Lorsqu'une demande avec un timestamp plus petit arrive une fois le temps de la mémoire tampon est terminé, le système subit un échec, parce que le message de quelqu'un d'autre a commencé à être envoyé aux auditeurs. Dans ce cas, en plus de couper l'utilisateur parlant, les auditeurs voient la collision parce qu'ils entendent deux messages différents.

Stratégies d'accès à la capacité par satellite

Dans un scénario tel que celui considéré dans cette thèse, un réseau cellulaire LTE temporaire qui est connecté réseau de cœur au travers d'une liaison satellite, le terminal satellite pourrait utiliser un lien satellite retour standard. Au lieu d'avoir une bande passante allouée fixe, un terminal satellite avec un débit de symboles faible utilise une bande passante qui est partagée entre plusieurs stations. Par conséquent, une utilisation efficace des ressources peut se traduire par une augmentation en termes du nombre d'utilisateurs et d'une meilleure performance globale. Fondamentalement, il existe trois façons d'accéder au lien retour ce scénario en compétition: accès fixe (FAMA), accès à la demande (DAMA) et l'accès aléatoire (RA). Le trafic Push-to-Talk comprend deux catégories de trafic: la signalisation et les trames de voix. La signalisation transmise via le lien satellite comprend les demandes et les relâches du floor. En outre, les messages de refuse et d'acceptation pourraient également être transmis alors qu'il ne sont pas nécessaire, étant donné le protocole de floor control distribué décrit auparavant. Les paquets de voix ont un niveau de priorité élevé, similaire à celui des appels vocaux réguliers. Les trames de signalisation sont des paquets courts qui nécessitent un faible retard et une livraison garantie. Il est préférable qu'ils soient traités avec la capacité d'accès fixe si possible. En outre, le canal d'accès aléatoire peut également être utilisé dans le cas où la congestion est limitée et la probabilité de succès reste élevée. Compte tenu de cette approche, le retard inhérent d'allocation pourrait être évité et le réseau de cœur serait rapidement informé des demandes entrantes. Les paquets PTT vocaux sont de nature différente. Ils se composent de trames vocales codées à un intervalle constant de quelques dizaines

RESTRICTED DOCUMENT

de millisecondes, généralement de 20 ms. Les appels ont généralement une courte durée et il existe une période sans transmission entre les différents appels. Plusieurs appels forment une conversation, qui peut être reconnue parce que le temps moyen entre deux conversations est nettement plus long que celui entre deux appels de la même conversation. Les paquets de voix sont encore courts par rapport à d'autres services et ils se composent généralement d'un en-tête comprenant IP / UDP / RTP de 40B et une charge utile de 20B-60B selon le protocole de codage. L'utilisation de l'allocation fixe pour accueillir la voix PTT serait parfaite du point de vue de la QoS. Pourtant, le trafic PTT est plus court et moins constant par rapport aux autres services VoIP. Par conséquent, cette allocation fixe pourrait être surestimée et les ressources perdues dans les cas où la station manipule uniquement du trafic PTT. Lors que ce n'est pas possible d'utiliser une allocation fixe, il est important que le retard de transmission de la demande et le délai initial de l'appel soit similaire. Dans le cas contraire, on pourrait avoir des problèmes d'inconsistance. Quand une demande de floor est envoyée par l'intermédiaire des ressources de l'allocation fixe ou en utilisant le canal d'accès aléatoire et, d'autre part, une demande de capacité en débit est délivré pour gérer les trames de voix, le retard global d'allocation pour ces dernières rend inutile le mécanisme de mise en mémoire tampon introduit dans le protocole de floor control. Afin d'éviter ce scénario, le mieux est d'exploiter le canal d'accès aléatoire (RA) pour envoyer les premiers paquets de voix. On prévoit que la transmission devrait durer quelques secondes. Le canal RA peut être utilisé tout au long de l'appel dans le cas où la charge sur ce canal reste faible. Toutefois, pour augmenter la stabilité, il est préférable d'envoyer une demande de capacité avec les premières trames de voix afin d'obtenir des ressources dédiées. Par conséquent, les premiers paquets seront envoyés par RA jusqu'à ce que la confirmation de l'allocation soit reçue. Une demande de capacité en débit serait adaptée pour gérer la voix PTT jusqu'à sa fin. La demande de capacité pourrait également être envoyée en même temps de la requête du floor control, même si la probabilité qu'un autre utilisateur obtient le floor reste encore élevée. Si le détenteur final est un autre utilisateur géré pour la même station de retour, les ressources destinées au premier utilisateur seront données au titulaire final. Pour évaluer les diverses méthodes d'allocation, nous avons besoin de générer du trafic qui se rapproche du comportement d'un groupe PTT typique. Nous présentons un modèle de conversation PTT à partir du modèle VoIP ON-OFF classique et les résultats sur la caractérisation des conversations PTT dans la littérature existante. Par la suite, nous examinons certaines méthodes d'attribution: un taux déterminé pendant la durée de toute une conversation, une demande de débit pour chaque message et une combinaison de ceci avec soit une allocation minimale ou la disponibilité d'un mécanisme d'accès aléatoire. Les deux derniers donnent les meilleurs résultats en termes de retard moyen sur les files d'attente et de jitter grâce à la possibilité de transmettre des trames avant la confirmation d'une demande ait été reçue. Pourtant, l'attribution d'une capacité minimale est très inefficace étant donné que les groupes PTT restent non actifs la plupart du temps. Le trafic généré par un groupe PTT est sporadique, donc la combinaison d'une demande en débit et l'accès aléatoire est la meilleure approche.

RESTRICTED DOCUMENT

Optimisation de la transmission des données des utilisateurs

La mise en place des réseaux par paquets était une avancée majeure pour les communications et les services de transmission de la voix. L'adoption de la voix sur IP a permis à plusieurs utilisateurs de partager un canal commun et, par conséquent, d'augmenter le nombre de clients servis sans diminuer leur qualité d'expérience. Cependant, l'envoi de la voix sous forme de paquets pose des nouvelles questions. Chaque trame vocale qui est mise en paquets est précédée d'un en-tête. Les paquets de voix sont généralement de petite taille par rapport à d'autres types d'information, mais l'en-tête lui-même reste similaire. Par conséquent, la partie du paquet qui contient les informations d'en-tête peut être plus grande que celle qui comprend la voix. Ceci affecte le débit final de la liaison, ce qui réduit le nombre de flux ou d'utilisateurs qui peuvent être servis en même temps. En règle générale, la voix est envoyée avec un en-tête RTP (12 octets), un en-tête UDP (8 octets) et en-tête IP (20 octets pour IPv4 ou 60 octets pour IPv6). La taille de la charge utile, en fonction du protocole de codage, peut être aussi faible que 12 octets (cas pour AMR avec 4,75 kbps de débit binaire). Les valeurs classiques se trouvent entre 20 et 160 octets, en fonction du débit binaire et la période de mise en paquets. Par conséquent, le coût final introduit par la série d'en-têtes peut être plus de deux fois la taille réelle de la charge utile. Enfin, il est important de rappeler que les applications VoIP envoient généralement 30-50 paquets par seconde, ce qui aggrave cet effet. Deux techniques principales sont utilisées pour réduire l'impact de cette surcharge: le multiplexage des trames et la compression des en-têtes. Le multiplexage regroupe plusieurs trames vocales, à savoir, des charges utiles, dans un paquet unique, réduisant le poids de l'en-tête sur la taille totale du paquet. D'autre part, la compression d'en-tête exploite la redondance de certains des champs et réduit l'ensemble des informations envoyées par paquets consécutifs. Notre approche est d'utiliser les deux techniques afin de réduire le nombre de bits nécessaires pour transmettre une session de VoIP, en essayant d'augmenter le nombre d'utilisateurs, tout en maintenant une bonne qualité de service. Nous voyons PTT voix comme une forme de VoIP, de sorte que nous cherchons à appliquer les mêmes techniques, avec quelques différences, car ces services sont susceptibles d'être utilisés au même temps, dans une cellule LTE déployée reliée à une liaison satellite. Les mécanismes de multiplexage typiques créent des paquets qui incluent des mini en-têtes de taille constante en plus des en-têtes RTP complets. Ces mini en-têtes comprennent un numéro d'identification de contexte qui permet au récepteur de classer correctement la trame et la transformer en un paquet régulier. Cependant, il est prévu que, lorsqu'une nouvelle transmission commence, l'initialisation du contexte doive être traitée de façon séparée. Dans le cas rare où certains champs des en-têtes IP-UDP changent, ils doivent être mis à jour d'une manière similaire. Si nous considérons que l'ID de contexte est d'au moins 1 octet, le mini en-tête serait en quelque sorte plus large et si on lui ajoute au même temps un en-tête RTP, le multiplexeur a besoin d'envoyer de l'ordre de 10 octets pour chaque trame. Au lieu d'utiliser des mini en-têtes et des en-têtes RTP complets, nous proposons d'utiliser des en-têtes compressés, tout comme avec RoHC (Robust Header Compression). La plupart des trames peut être envoyée avec un en-tête comprimé de seulement 2 octets. En plus, l'initialisation et mise à jour des en-têtes peuvent être envoyés directement dans le paquet de MUX comme toute autre

RESTRICTED DOCUMENT

trame. D'ailleurs, nous considérons que certains des champs des en-têtes IP ne sont pas utiles et donc les en-têtes d'initialisation peuvent être réduits en taille. La seule question qui est apparu au début était que RoHC a besoin d'une autre couche supérieure qui fournit la taille de la trame. Toutefois, compte tenu d'une application VoIP, la taille de la charge utile change rarement et, par conséquent, nous pouvons la déduire de la partie d'en-tête IP de l'en-tête d'initialisation et de mise à jour. L'opération de décompression est alors le même que dans RoHC. Le décompresseur analyse trame par trame pour récupérer le paquet d'origine. Il est possible qu'il ne soit pas capable de décompresser une trame. Il doit alors la jeter et passer à la suivante. Le décompresseur connaît les dimensions typiques des en-têtes et des charges utiles, donc il peut tenter de décompresser la trame suivante. Nous prévoyons que le nombre possible de tailles de charge utile, c'est-à-dire les taux de l'application VoIP, soit limité. Par conséquent, une seule erreur ne signifie pas jeter le reste du paquet. La compression suit à nouveau la méthode d'opération de RoHC, en changeant son état de compression, tout comme dans le mode unidirectionnel. Il est possible de compresser les en-têtes externes également, ceux du paquet MUX, dans le terminal satellite si la paire de passerelle-terminal dispose d'un mécanisme de compresseur-décompresseur. La raison pour laquelle nous décidons de faire un paquet de MUX avec des en-têtes comprimés est d'éviter la compression-décompression saut par saut. Par conséquent, la compression est effectuée une fois dans le premier super nœud et continue comme cela jusqu'à ce qu'il arrive au super nœud de destination. La combinaison du multiplexage et de la compression d'en-tête est très prometteuse et pourrait se traduire par une augmentation du nombre de sessions desservies. Nos simulations montrent un potentiel pour presque doubler cette métrique.

La différenciation de service entre VoIP et PTT

La solution que nous avons présentée dans le chapitre précédent considère un traitement similaire des appels VoIP et messages PTT . Les deux sont des services vocaux qui sont considérés comme premium dans le scénario de réseaux cellulaires déployés temporairement. Cependant, il existe une petite différence : le PTT est unidirectionnel. Comme il n'y a pas besoin de se synchroniser avec un flux qui vient dans le sens opposé, le PTT fonctionne bien même avec un plus grand retard, parce que le message peut être lu correctement. Par conséquent, il est possible d'augmenter le nombre de sessions simultanées encore plus qu'en utilisant seulement la compression d'en-tête et le multiplexage, en appliquant une différenciation de service par retard entre ces applications. Le but est de permettre un retard de multiplexage supérieur pour l'application PTT, tout en conservant le délai des appels VoIP aussi bas que possible. Un rapport de différenciation peut être ensuite calculé en divisant l'objectif de délai des deux services. Nous croyons qu'avoir une différenciation parfaite serait difficile compte tenu de l'attente inhérente entre chaque opportunité de transmission dans le système satellite. En outre, on doit considérer que plusieurs trames partent en même temps dans le même paquet MUX. Notez que le taux d'erreur pour délai excessif des deux applications devrait être similaire car ils ont les mêmes exigences en ce qui concerne la robustesse contre la perte de trame. Nous examinons trois différentes possibilités d'effectuer

RESTRICTED DOCUMENT

une telle différenciation. La première est fondée sur le Deficit Weighted Round Robin (DWRR). DWRR divise le trafic en N types et donne à chaque type ou classe un quantum qui représente le nombre total d'octets qu'une classe devrait envoyer à chaque tour. Dans le cas où certains octets ne sont pas utilisés, ils sont décalés au tour suivant. Par conséquent, chaque classe est donnée une partie ou pourcentage de la capacité de sortie qui est égale au rapport entre le quantum de la classe et la somme des quanta de toutes les types. Nous classifions les trames entre la VoIP et le PTT afin de les mettre sur deux queues FIFO séparées. Il est possible d'obtenir la différenciation de retard souhaité et de limiter au maximum retard de la file d'attente en choisissant correctement le poids et la taille de chaque file d'attente. La complexité de cette mise en œuvre de l'ordonnanceur est simple mais la configuration des paramètres est le principal défi. En effet, nous observons que ce mécanisme exige beaucoup de réglage fin et a des difficultés pour maintenir le délai maximum. Pour cette raison, nous avons décidé de prendre le percentile 99e comme métrique. La deuxième solution que nous présentons est une adaptation d'un programmateur horodaté qui s'appelle Earliest Due Date (EDD). Elle attribue la priorité de transmission basée sur des dates limite. Les trames ne sont plus divisées entre la VoIP et PTT. Au lieu de cela, une seule file d'attente, ordonnée par date limite, est utilisée. Les trames avec des délais limite plus courts seront situées au début de la file d'attente et, par conséquent, elles seront les premières servies. Lorsqu'un paquet MUX est envoyé, le protocole supprime les trames à l'intérieur de la file d'attente qui étaient incapables de répondre à leur échéance. Cette politique impose un délai strict. Les charges de traitement augmentent en raison du besoin de réordonner systématiquement les trames dans la file d'attente, mais la configuration est simple. En donnant à chaque service un délai limite différent et avec le retrait des paquets quand ils le dépassent, l'ordonnanceur fait un bon travail en limitant le retard et fournisse la différenciation de service. Toutefois, le taux d'erreur pour délai excessif des deux services n'est pas cohérent et plus d'analyse devrait être mené avant de l'inclure dans un système réel. Le dernier mécanisme, Hybrid Proportional Delays (HPD) a été proposé dans le cadre du Proportional Differentiated Services. Les auteurs suggèrent une série d'ordonnanceurs pour effectuer la différenciation des services par retard. Leur objectif est d'effectuer une différenciation par rapport continu, en cherchant à avoir un rapport de retard égal au rapport des paramètres de différenciation. Nous considérons leur dernière méthode proposée qui vise à avoir un la différenciation proportionnelle de retard sur le long terme, en moyenne, et à court terme, instantanément. Dans notre scénario, le trafic est divisé en deux files d'attente FIFO. A chaque occasion de transmission, le protocole sélectionne le service avec le "retard hybride normalisé" maximal. Ce paramètre consiste en une composante du retard moyen et un autre de retard de file d'attente instantanée du premier paquet de la file d'attente. Au lieu d'avoir une taille de file d'attente pour chaque type, il existe une taille globale unique. Lorsque la file d'attente est pleine, une trame du service avec le taux d'échec inférieur est sélectionnée. De cette façon, le but est d'égaliser les taux de ces deux services. La complexité de ce mécanisme est élevée. Les trames doivent être horodatées et le retard moyen de la file d'attente est constamment mis à jour. En outre, le taux d'échec doit être calculé ainsi à chaque décision. Avec ce dernier mécanisme, nous avons aussi des problèmes pour contrôler le retard maximal et nous choisissions le percentile 99e une nouvelle fois. Les résultats d'opération sont très bons et nous obtenons

RESTRICTED DOCUMENT

le même taux d'échec pour la VoIP et le PTT. Néanmoins, il nécessite d'une grande charge de calcul, car il a besoin de revoir le délai d'attente du premier paquet de la file d'attente et de recalculer le taux d'échec en continu.

Adaptation aux changements du trafic

L'idée de ce dernier chapitre est d'intégrer la technique d'accès par satellite basé sur la demande d'un taux déterminé et d'un canal d'accès aléatoire avec l'ordonnanceur Earliest Due Date présenté dans le chapitre précédent. Nous simulons des groupes PTT et des appels VoIP en provenant de la cellule LTE par satellite et par conséquent, qui utilisent le lien retour du satellite. En ce qui concerne le PTT, nous avons considéré que l'utilisateur transmet à tout moment, pendant tout le message, c'est-à-dire sans la détection d'activité vocale. En outre, certains des appels PTT peuvent venir de l'autre côté du satellite, ce qui réduit effectivement la charge sur le côté de retour. Pour la VoIP, le modèle d'activité vocale classique a été pris, semblable aux chapitres précédents. Quand un nouvel appel commence, le terminal satellite émet une demande pour la capacité totale de l'appel, ne considérant pas la possibilité de se bénéficier de la capacité supplémentaire disponible à partir d'autres appels lorsque les utilisateurs ne parlent pas. Compte tenu des paramètres du système similaires à ceux pris pendant la comparaison des techniques d'accès par satellite, nous observons que le système fonctionne bien en général. Cependant, il y a deux principaux problèmes qui font trébucher la solution: (a) la surcharge, étant donné que, sans contrôle d'admission tous les appels / messages sont acceptés, et (b) l'arrivée soudaine de multiples appels, ce qui augmente la charge pendant une période courte de temps avant la mise à jour de l'allocation de ressources. Ce chapitre se concentre sur ce dernier problème. Une grande charge venant des appels VoIP réduit le problème étant donné que normalement le système aura une certaine capacité de réserve. Par conséquent, nous mettons l'accent sur les cas dans lesquels la charge des messages en provenance de PTT est plus importante. Les problèmes potentiels que nous avons identifiés se produisent lorsque plusieurs appels arrivent dans le système dans une courte période de temps. Dans notre exemple, il y a une possibilité de demande de ressources tous les 270 ms (équivalent au retard à sens unique de la liaison par satellite), qui se traduit par un retard de mise à jour avec un délai d'aller-retour. Le délai d'attribution de nouvelles ressources peut impliquer que pendant certaines périodes, le système soit surchargé. Cette situation peut provoquer des pertes de plusieurs paquets à cause de délais excessifs au début des nouveaux appels qui pourraient empêcher l'auditeur de comprendre le début du message. Ces erreurs sur quelques paquets consécutifs sont ce qui détériore le plus la qualité d'expérience, en particulier quand elles se produisent au début du message. En outre, nous ne considérerons pas ici d'autres causes d'erreurs, telles que celles correspondantes à la couche physique du réseau mobile (à la transmission ou à la réception) ou sur la liaison par satellite, ce qui se produit généralement en rafales. Par conséquent, notre intérêt ici est de réduire les situations où plusieurs paquets sont perdus en raison de trop de retard. Pour surmonter ce problème, nous proposons deux méthodes de prévention pour gérer les sessions PTT qui arrivent dans les périodes de pointe. Dans le mode dégradé, le délai de la file d'attente pour ces sessions peut

RESTRICTED DOCUMENT

augmenter jusqu'à 200 ms afin de prioriser les sessions déjà en transmission. A son tour, dans le mode de retard de démarrage, nous proposons de retarder le début de la transmission jusqu'à ce que de nouvelles ressources soient attribuées. Les deux méthodes sont suffisantes pour réduire la présence d'erreurs sur plusieurs paquets consécutifs, mais il est plus difficile d'évaluer le délai de démarrage réel de celui-ci, parce qu'il résulte très variable d'une session à l'autre lors que le numéro de groupes PTT simulés augmente.

RESTRICTED DOCUMENT

Contents

RESTRICTED DOCUMENT

RESTRICTED DOCUMENT

RESTRICTED DOCUMENT

List of Figures

RESTRICTED DOCUMENT

RESTRICTED DOCUMENT

RESTRICTED DOCUMENT

List of Tables

Glossary

3GPP	3rd Generation Partnership Project
ACK	Acknowledgement
ACM	Adaptive Coding and Modulation
AF	Assured Forwarding
API	Application Programming Interface
Airbus DS	Airbus Defence and Space
AMR	Adaptive Multi-Rate
AMR-WB	Adaptive Multi-Rate Wideband
APCO	Association of Public-Safety Communication Officials
AS	Application Server
BE	Best Effort
BER	Bit Error Rate
BM-SC	Broadcast Multicast Service Center
BoD	Bandwidth on Demand
BT-DAMA	Burst Targeted DAMA
BTS	Base Transceiver Station
CBR	Constant Bit Rate
CDMA	Code Division Multiple Access
CR	Capacity Request
CRA	Constant Rate Assignment
CRC	Cyclic Redundant Code
CRDSA	Contention Resolution Diversity Slotted Aloha
CSMA	Carrier Sense Multiple Access
DA-AC	Dedicated Access - Allocation Channel
DAMA	Demand Assignment Multiple Access
DEMUX	Demultiplexor
DiffServ	Differentiated Services
DMO	Direct Mode Operation
DMR	Digital Mobile Radio

DQPSK	Differentially encoded Quadrature Phase-Shift Keying
DRR	Deficit Round Robin
DSA	Diversity Slotted Aloha
DVB	Digital Video Broadcasting
DVB-S	Digital Video Broadcasting – Satellite
DVB-RCS	Digital Video Broadcasting – Return Channel Satellite
DWRR	Deficit Weighted Round Robin
ECC	Electronic Communication Committee
EDD	Earliest-Due-Date
EDF	Earliest Deadline First
EDGE	Enhanced Data rates for GSM Evolution
eMBMS	Evolved Multimedia Broadcast/Multicast Service
eNB	evolved – Node B (LTE base station)
E-UTRA	Evolved Universal Terrestrial Radio Access
EF	Expedited Forwarding
EPC	Evolved Packet Core
EPS	Evolved Packet System
ETSI	European Telecommunications Standards Institute
FC	Floor Control
FCA	Free Capacity Assignment
FCC	US Federal Communications Commission
FCFS	First Come First Serve
FAMA	Fixed-Assignment Multiple Access
FDMA	Frequency Division Multiple Access
FIFO	First In First Out
FirstNet	First Responder Network Authority
FM	Frequency Modulation
FR	First Responders
GCS AS	Group Communication Service Application Server
GCSE	Group Communications Service Enabler
GEO	Geostationary Earth Orbit
GERAN	GSM EDGE Radio Access Network
GMSK	Gaussian Minimum Shift Keying
GPRS	General Packet Radio Service
GPS	Global Positioning System
GSM	Global System for Mobile communications
GSMA	Groupe Speciale Mobile Association
GW	Gateway

RESTRICTED DOCUMENT

HARQ	Hybrid Automatic Repeat request
HD	High Definition
HPA	High Performance Amplifier
HPD	Hybrid Proportional Delays
HTTP	HyperText Transfer Protocol
IAT	Inter Arrival Time
IEEE	Institute of Electrical and Electronics Engineers
IMS	IP Multimedia Subsystem
IOPS	Isolated E-UTRAN Operation for PS
IP	Internet Protocol
ISI	Inter Symbol Interference
ITU	International Telecommunication Union
LEO	Low Earth Orbit
LSB	Least Significant Bit
LTE	Long Term Evolution
MAC	Media Access Control
MCE	Multi-cell/Multicast Coordination Entity
MCPTT	Mission Critical PTT for LTE
MEO	Medium Earth Orbit
MIMO	Multiple-Input Multiple-Output
MME	Mobility Management Entity
ModCod	Modulation and Coding
MSS	Mobile Satellite Systems
MTU	Maximum Transmission Unit
MUX	Multiplexor
MuSe	Multipoint Service
NACK	Negative Acknowledgement
NCC	Network Control Center
NPSTC	National Public Safety Telecommunications Council
NTIA	National Telecommunications and Information Administration
OBP	On-Board Processor
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
OMA	Open Mobile Alliance
OTT	Over-The-Top
P25	Project 25
PAPR	Peak to Average Power Ratio
PDCP	Packet Data Convergence Protocol

RESTRICTED DOCUMENT

PDF	Probability Density Function
PDN	Packet Data Network
PDS	Proportional Differentiated Services
PEP	Performance Enhanced Proxy
P-GW	Packet Data Network Gateway
PLFRAME	Physical Layer Frame
PLMN	Public Land Mobile Network
PMR	Professional Mobile Radio
PoC	Push-To-Talk over Cellular
PPDR	Public Protection and Disaster Relief
ProSe	Proximity Services
PS	Public Safety
PSK	Phase-Shift Keying
PTT	Push-To-Talk
QAM	Quadrature Amplitude Modulation
QoE	Quality of Experience
QoS	Quality of Service
RA	Random Access
RA-AC	Random Access Allocation Channel
RBDC	Rate Based Dynamic Capacity
RC	Request Class
RCST	Return Channel Satellite Terminal
RLC	Radio Link Control
RLE	Return Link Encapsulation
RoHC	RObust Header Compression
RoIP	Radio over Internet Protocol
RRC	Radio Resource Control
RTP	Real-time Transport Protocol
RTCP	RTP Control Protocol
RTO	Retransmission Time Out
RTT	Round Trip Time
SA	Slotted Aloha
SACK	Selective Acknowledgement
SAE	System Architecture Evolution
SAT	Satellite
SIP	Session Initiation Protocol
SC-FDMA	Single Carrier Frequency Division Multiple Access
SCTP	Stream Control Transmission Protocol

RESTRICTED DOCUMENT

S-GW	Serving Gateway
SN	Super Node
SNIR	Signal-to-Noise plus Interference Ratio
SSA	Spread Spectrum Aloha
TBTP	Terminal Burst Time Plan
TCP	Transmission Control Protocol
TDMA	Time Division Multiple Access
TEDS	TETRA Enhanced Data Service
TETRA	Trans European Trunked Radio
TMGI	Temporary Mobile Group Identity
TS	Time Stamp
TTI	Transmission Time Interval
TWTA	Traveling-Wave Tube
UDP	User Datagram Protocol
UE	User Equipment
UTRAN	Universal Terrestrial Radio Access Network
VBDC	Volume Based Dynamic Capacity
VBR	Variable Bit Rate
VoIP	Voice over IP
VoLTE	Voice over LTE
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network

RESTRICTED DOCUMENT

Chapter 1

Introduction

1.1 Background and Motivation

1.1.1 Context

Push-To-Talk (PTT) is a simple communication service that allows a group of people to exchange voice messages. The concept is self-explanatory: the user pushes a button to request the permission to speak and if granted, it talks and the service delivers the message to the rest of participants. The communication is half-duplex, one-way, in opposite to full-duplex services such as regular voice calls. By enforcing that only one person speaks at a time and using procedure words, users can communicate effectively. PTT is still the main way of communication among law enforcement agents and other emergency services.

In the recent decades, Public Safety agencies have been provided with dedicated networks that feature services such as PTT. These solutions are commonly referred as Professional Mobile Radio (PMR) and differentiated from Public Land Mobile Network in terms of requirements: reliability, availability, robustness... The technology evolution and the associated costs of dedicated networks have risen the interest in searching for new solutions merging with other commercial networks. With the deployment of Long Term Evolution (LTE), mobile networks have adopted a full IP architecture. PMR-like functions and requirements are now introduced in the new versions of the standard.

1.1.2 Purpose of this work

Mobile and Internet networks provide the infrastructure necessary to allow millions of people to communicate around the globe. However, as any other infrastructure on earth, they are exposed to unexpected events that may damage them and become unavailable. Earthquakes, floods, power outages, human sabotages or simply over-demand could disable such networks and jeopardize the communications. In life-threatening situations, communications services are indispensable and temporary solutions need to be deployed. Satellite services can rapidly provide a global coverage at any time and enable a faster response.

Airbus Defence and Space commercializes a solution that allows deploying a LTE network and use

a satellite link to connect with the rest of the network towards the Internet. Given that Public Safety agencies are the main customer of such solution, it became necessary to incorporate a PTT service.

Our objective is to optimize the mechanisms conceived for these services in the regulated terrestrial world and adapt them to the characteristics of the satellite segment and its interconnection with the rest of the network. The satellite interconnection challenges the current PTT systems because of the inherent delay and resources scarcity.

The communication between users at both sides of the satellite link poses first a problem in the group management, hence it is necessary to modify the existing protocols in order to effectively consider the different access latencies. Furthermore, the traffic on the return side of the satellite system has to be used wisely. The use of capacity demand mechanisms (DAMA) improves the efficiency at the same time that limits the capacity of real-time applications. It is expected that other services, such as video streaming for monitoring, increase in importance in the following years. Therefore, it becomes crucial to reduce the bandwidth needs for voice services.

The purpose of this work is to design a PTT solution that builds upon two main components: a control plane that takes into account the delay of the satellite link and a optimal transmission of multiple voice streams in the user plane.

1.2 Push To Talk

PTT is a simple communication service that allows two or more users to exchange voice messages in real-time. From the times of analog communications, the idea of PTT was really straightforward: by pushing a button on the user terminal, one can start emitting in a given frequency, and the rest of terminals, granted they are configured in the same band, receive the message while their button is not pressed. In this way, a half-duplex communication is created.

PTT has since been the service of choice in environments where coordination and logistics are primordial: emergency services, public safety (PS), delivery services, factory operations... The capacity of enabling instant group communications is greatly appreciated in this situations. Plus the certainty of having only one user speaking fulfills perfectly the operation protocol of such organizations.

Analog device-to-device technologies evolved to digital infrastructure-based solutions, mainly used for the first two types of users cited previously. States around the world have deployed their own nation-wide networks of what we call Private Mobile Radio (PMR) networks. Multiple systems such as P25, TETRA or TETRAPOL have been standardized.

The arrival of smart phones raised the necessity of a solution that could be accessed from a regular mobile terminal instead of the proprietary, and often expensive, PMR equipment. In that sense, PTT over Cellular (PoC), a system based on the IP Multimedia Subsystem (IMS), was proposed. Several carriers launched a PTT service based on this technology for their GPRS or CDMA networks. They rapidly received the attention of multiple organizations but PS continued relying on their dedicated infrastructure.

The evolution of the mobile networks, the increasing level of obsolescence of the legacy PMR sys-

RESTRICTED DOCUMENT

tems and the growing demand for other data services pushed the decision to adopt Long Term Evolution (LTE) as the technological driver for future PMR solutions.

PTT over Satellite One of the reasons to start this work was the limited previous work about PTT services over satellite. No solution was initially conceived to work over satellite links. However, their capability to increase coverage to almost all the globe remains unrivaled by the previously cited technologies. Hence, as we are going to review in the next chapter, some of these systems have been tested with a satellite backhaul link without designing a specific adaptation.

Some of the Mobile Satellite Systems (MSS) currently in-orbit feature PTT over Satellite. Yet, they still rely on a single technology, Iridium for example, which translates into a limited terminal offer. This also rises an interoperability problem, increasing the difficulty of coordinating multiple organizations or users without a satellite terminal. Furthermore, they impose the use of a satellite link, while some communications could be managed by temporary terrestrial infrastructures covering a given zone.

The solutions adopted in this work rely on typical cellphones and remain interoperable with all systems using the Internet. This allows reducing the user part complexity and a smart use of the satellite resources.

1.3 Contributions of this work

1.3.1 Main Contributions

This thesis can be divided in two main complementary parts:

A Distributed Floor Control Protocol The first part of this work focus on the control part of the PTT service, and more specifically on the definition of an optimal floor control protocol. Floor control refers to the set of mechanisms that allow managing the permission to speak in a group communication service based on PTT. The communication between users on both sides of a satellite link challenges the existing systems, mainly based on centralized approaches.

First, we analyze the possible architecture choices advancing towards a distributed solution. Then, the concept of an optimistic floor control is introduced. It allows reducing the access time for the users requesting to speak and, by establishing a buffering process, we increase the fairness among the users independently of their location. The major part of the first year was dedicated to the definition and validation of the probabilistic analytical model that supports this invention.

This protocol will be integrated with existing PTT applications in the following months under the supervision of Airbus DS in order to be tested in real scenarios.

Optimization of the User Plane The second part of this dissertation was devoted to the analysis of the distributed access methods and the integration of different mechanisms in order to improve the Quality of Service (QoS).

RESTRICTED DOCUMENT

We first reviewed the diverse demand assisted multiple access techniques that allow sharing the resources of the satellite return link among different users. We analyzed the traffic dynamics of a typical PTT group communication and compared the different resource allocation approaches.

After this study, we focused on the optimization of the resources used for voice services. From the beginning, we realized that there was an opportunity to tackle this issue together for common voice calls, VoIP, and PTT given the similarities of their packet format. When transmitting multiple flows, it is possible to multiplex them and reduce the proportion of the resources used to convey the header information. Then, we integrated the RObust Header Compression (RoHC) framework in order to decrease the number of bits transmitted in the headers. Consequently, we demonstrated that the union of these mechanisms allowed doubling the number of served sessions.

Following the approach of optimizing VoIP and PTT together, we proposed to apply delay differentiation between the two services. Given that PTT is unidirectional, there is a chance to adapt the queuing delay depending on the system load. We evaluated three scheduling possibilities and proved the potentiality to increase the capacity of the system even further.

Finally, we performed a last study combining one of the resource demand mechanisms and one scheduling protocol and observed the behavior. We confirmed the overall performance and observed how the arrival of multiple calls in a short period of time challenges the resource allocation and proposed two preventive mechanisms avoiding burst of dropped packets at the beginning of new messages.

1.3.2 Publications, Patents and Related Documents

The work presented in this document has resulted in one technical report, two conference papers and one patent. [12] discusses the different architecture possibilities in order to provide PMR services on LTE. [13] presents the distributed floor control protocol designed during the first part of this thesis and it was presented during the IEEE Global Humanitarian Technology Conference of october 2014. A patent protecting this invention [14] was filled in the French patent office on behalf of Airbus Defence and Space. Finally, on its turn, [15] discusses the framework integrating multiplexing and header compression as well as it presents and compares three different scheduling mechanisms to provide delay service differentiation between VoIP and PTT. This last paper was presented in the 21th KA and Broadband Conference in october 2015.

1.4 Outline of this document

The rest of this manuscript is organized as follows:

Chapter II provides an extensive background about the main topics treated in this thesis. We discuss the technologies involved and the historical evolution of Public Safety communications and Private Mobile Radio more specifically. In addition, an introduction to the LTE standard is provided. We analyze the advantages and challenges of adopting LTE as the main technology for PS. Finally, we review the shortcomings of the satellite environment and some of the protocol adaptations that have been proposed.

RESTRICTED DOCUMENT

After the introductory chapter, chapter III discusses the diverse architectures considered and the final choice in order to provide a PTT service over hybrid networks concerning LTE and Satellite.

Chapter IV presents the floor control protocol. First, we introduce the subject of such management mechanisms and analyze the previous work on similar scenarios. Then, we discuss about the approach adopted and present the working principle of the protocol. After, we deepen into the analytical model supporting our design. To conclude, we present the results of an important evaluation study validating our proposal.

Chapter V provides some insight about the different resource demand methods available for the return link of a satellite system and analyze which are the most promising options regarding emergency communication services such as PTT. We take this opportunity in order to discuss about the behavior of a typical PTT group and the traffic it generates.

In chapter VI, we integrate some multiplexing and header compressing techniques in order to reduce significantly the necessary data rate dedicated to one voice flow. We describe the motivation and rationale behind the definition of the Robust Header Compression framework and discuss about the header formats and characteristics.

Chapter VII compares three scheduling mechanisms that are configured to provide delay service differentiation between VoIP and PTT. We also provide some background about the main scheduling protocol families before detailing our choices and comparing them through a simulation model built upon the one introduced in the precedent chapter.

Chapter VIII combines one of the access methods presented in chapter V and one of the scheduling protocols proposed in the previous chapter. We analyze the possible challenges that may occur during the service of multiple PTT groups and voice calls and how we can reduce their impact. We observe some particular cases and compare two proposed prevention methods. Finally, some broader simulations are carried out and we compare the outcomes with the baseline scenarios.

Finally, in chapter IX, we discuss about the main outcomes of this dissertation and we provide an outlook to the possible future work.

RESTRICTED DOCUMENT

Chapter 2

Background

2.1 Public Safety Communications

2.1.1 Public Safety Context

Public Safety (PS) comprises the work of multiple organizations or agencies who are engaged in the protection of people, the environment and other general assets. They face a variety of challenges including natural incidents, terrorism, transportation accidents, medical urgencies, search and rescue, and other natural or human caused threats.

Therefore, examples of PS organizations are the police, the fire-fighters, field medical responders, transportation security guards, border or costal guards... Military may also be included in the definition; while they maintain a differentiated status, they collaborate with the previous bodies in several scenarios.

[16] provides an extensive introduction to the different functions of the PS agencies, their communications needs and the available solutions. In this section, we summarize the operational context, the requirements regarding communications, the related radio frequency regulations and the solutions designed to benefit from the satellite resources.

2.1.2 Operational scenarios

Generally, the situations in which PS organizations participate can be divided into two main blocks: routine operations, including the activities and issues they face in a daily basis, and disaster relief situations or events where a large number of people or a great area is impacted. The latter are more uncommon and they frequently need the intervention of multiple agencies. Furthermore, in this scenario, it is more likely that critical infrastructures, such as energy, communications or transportation have been impacted, challenging the work of the first responders.

The location of the scenarios can be classified in urban environments, border areas, transportation infrastructures (ports, airports, roads, railways...) or rural environments.

The main operational scenarios requiring an important support from PS agencies are: emergency crisis in urban areas, such as fires or terrorist attacks; large natural disasters, like flooding or earthquakes;

cross-border operations, derived from threats like illegal immigration, smuggling or cross-national political tensions; and major events, like sports competitions, concerts or demonstrations.

2.1.3 PS Communications Requirements

The capacity of efficiently exchange information is crucial for PS to correctly face the scenarios described above. The communications systems used by these agencies are usually referred as Professional Mobile Radio (PMR). The communications requirements of these types of users are more restrictive than the ones of regular people. The correct operation of the communication system may help reducing the impact of the emergency crisis and aid first responders with their duties.

We first review the main services used by PS and then we discuss about the different requirements regarding the performance and the availability of those services.

The basic elements needed for PS communications are:

- *Voice:* This is the main mean of communication between first responders. It can be in form of one-to-one, also referred as private, calls or group calls, where a multiple users belonging or not to the same organizations are allowed to participate. Group calls are usually arbitrated in the form of Push-To-Talk solutions, where only one person is allowed to speak at a given time. PTT can also be used in private calls. Non-interactive voice communication like announcements, i.e. one-way and with no response required, are part of usual operations.

- *Data exchange:* It comprises a large number of services that require data exchange between multiple parties. These include video streaming, image or file exchange, access to maps or plans with more detailed information, database checks, sensor or biometric data monitoring, video surveillance...

- *Messaging:* The exchange of text messages is also an efficient way that requires less communications capabilities. This service is usually used as non-interactive and less urgent.

- *Security services:* PS needs access to security services regarding the confidentiality and integrity of their communications. Additionally, users have to be correctly authenticated and authorized to participate.

- *Location services:* This includes the tracking and monitoring of first responders and vehicles on duty. It can help to coordinate the different units or prevent them if something has occurred nearby.

The systems designed for the use of PS agencies, as PMR solutions, have more strict requirements compared to public commercial networks. Several reports provide detailed information of the service requirements and technical performance of these communication solutions [17] [18] . We review the main issues:

RESTRICTED DOCUMENT

- *Speech transmission performance:* Voice communication must guarantee the quality level to maintain the complete understanding of the messages even in situations with presence of background noise.
- *Quality of Service:* The main performance indicators that define a good quality of service are packet loss, latency (end-to-end delay) and jitter (delay difference between packets or frames). Performance requirements depend on the application at use. For example, voice communications require a low delay and jitter, while some packet loss may be acceptable (PS level of acceptance is more strict than regular voice exchange). Video streaming allows a higher latency but also requires a low jitter. File exchange relaxes the delay constrain but needs a very low packet loss. Data rate is also a key performance parameter, a high throughput is an increasing demand specially in data exchange applications such as video streaming or high resolution mapping.
- *Timeliness:* It is related to the expected quality of time. Real-time communications, or interactive, require a very low delay to maintain the quality of the information, audio or video, but also because they may need the immediate action of the receiving responders. Other non real-time services, however, are not that strict. Nevertheless, given the critical scenarios where users operate, the performance of data exchanged cannot be dismissed.
- *Prioritization:* The system may provide a mean to prioritize the calls in congested situations. PS organizations have a clear hierarchical structure and high level commands may override regular communications. Additionally, emergency calls, in situations where the life of a person may be in danger, have strict priority over other exchanges.
- *Robustness of the equipment:* Responders usually operate in difficult environments and their equipment should have a higher resistance to extreme temperatures, liquids and/or accidental drops.
- *Energy consumption:* Operations may last for hours and the availability of external power sources may be difficult in critical scenarios. Therefore, a low consumption is necessary to extend the operational life of the equipment.
- *Security:* As stated before, security is of primary importance as sensitive information is exchanged between responders.
- *Radio coverage:* The coverage of PS communications systems needs to be practically ubiquitous. While most of the systems are infrastructure based, it is possible to extend the coverage through the use of temporary networks connected through satellite or the direct communication between terminals.
- *Resilience/availability of the networks:* It is necessary that networks are available even in high saturated scenarios. Additionally, distributed processing and/or redundancy of infrastructure or core elements may be necessary. Finally, it has to be resistant to the external attacks, which may

RESTRICTED DOCUMENT

be physical, trying to destroy the equipment, or via cyber attacks, causing the improper functioning of the network.

- *Interoperability:* The possibility of the different systems to correctly work together has been a key demand from the PS community.
- *Scalability:* The possibility of the systems to continue to operation normally even with an increase on the number of active users or when covering larger areas.

Given the strict requirements listed above, PS agencies have generally demanded for dedicated networks. In that sense, the design of the network as well as its administrator is left to the organizations. This have been the trend for many years as agencies were reluctant to share the infrastructure with commercial operators. As it will be commented later, the possibility of using commercial networks for PS needs will come with the adoption of LTE and it will bring several advantages. Nevertheless, it remains clear that differentiated and dedicated spectrum for public protection and disaster relief is necessary to fulfill the cited requirements [19] . Commercial spectrum is expected to complement the performance of the dedicated resources.

2.1.4 Radio Frequency Spectrum Regulations

Public Safety organizations have demanded for dedicated network resources to effectively control their performances. With this goal, spectrum regulators allocate spectrum bands reserved for PS purposes and separated from the commercial portion of the frequency domain. The respective allocated bands depend on the studied region and there is no harmonization across countries.

Traditionally, reserved bands were located in the lower part of the spectrum under the 1 GHz mark where they could take advantage of the better radio propagation characteristics. Most of the PMR systems have been designed to be used in these conditions. However, the adoption of LTE opens the discussion in order to free new resources for PS services that need more bandwidth for broadband applications. The actual reserved bands may be limited in disaster relief scenarios with a large number of users. In that case, it could be interesting that PMR could use commercial spectrum using prioritization mechanisms.

In Europe, in 2008, the Electronic Communication Committee (ECC) decided to harmonize the frequency bands within the 380-470 MHz range [20]. The bands 380-385 MHz and 390-395 MHz have been designated for narrowband PMR systems. For the wideband systems, there is a higher flexibility and the specific channels are defined by the national authorities. Another ECC recommendation [21] recommends allowing the reserved use of 50 MHz in the 5 GHz band to boost broadband services. Additionally, the options of using the current TETRA and TETRAPOL bands, around 400-470 MHz, or freeing part of the spectrum used for TV broadcast, in the 694-790 MHz, are being discussed.

In USA, the spectrum allocation for legacy PMR systems is fragmented and it is a decision of the municipalities. This non harmonization has led PS organizations to increase their infrastructure resources to tap a wider spectrum, as the total reserved is larger than in Europe. This has been proved to be inefficient [22].

RESTRICTED DOCUMENT

In contrast with Europe, USA has taken very serious the adoption of LTE as the driving technology for broadband PMR systems. In 2012, The US Federal Communications Commission (FCC) together with the Congress have promoted the deployment of a nationwide LTE-based PMR network. This network has been named First Responder Network Authority (FirstNet) and it will be managed by an independent authority within the National Telecommunications and Information Administration (NTIA) [23]. 20 Mhz have been reserved in the 700 MHz range for the use of this network and 7 billion dollars will be invested. A dedicated standalone network has proved to be unfeasible and partnerships with commercial operators will be necessary.

2.1.5 PS Communications over Satellite

Satellite network are of great importance within the PS domain. Given their independence of terrestrial infrastructure and global coverage they can be used even in extreme scenarios. Additionally, temporary networks can be deployed in areas where the terrestrial resources are not available or are saturated and then be interconnected to the core network by means of satellite links.

The first satellite communications equipments of interest are the satellite phones that make use of the mobile satellite systems (MSS). In these systems, the terminal communicates directly with the satellite. While they provide a quick setup and are good for mobility, they are expensive, they have a limited transmission rate and it is not very efficient that all responders use them on the field. However, they provide a first mean of assistance at the beginning of the operation when temporary networks have not been yet deployed.

Then, fixed terminals provide higher data rates that could help to accommodate multimedia services. They can be used to interconnect the provisional deployed ad-hoc networks. The resulting network is often called hybrid, as it has two main components, the terrestrial part and the satellite link.

A common approach is to deploy a system based on a widely standardized technology, such as GSM or TETRA, and configure a backhaul through satellite. The backhaul is the portion the network between the access part and the core infrastructure. Some service providers feature fully operable GSM networks with satellite backhauling [24] [25] [26] [27]. Further, a report from the ETSI examines the concept of Emergency Communications Cells over Satellite (ECCS) [28] that relates to the idea recently reviewed.

Some studies have tried to interconnect PMR systems through satellite. For example, Cassidian tested the interconnection of a TETRA base station and a digital TETRA switch through satellite [29]. The connection proved to be successful even though it implied an increased delay. In [30], authors discussed about the implementation of the projected cell and the rest of the network. Basically, there are two architectures, the integrated one, where the base station connects with the switch through satellite, and the distributed one, where there are two core networks, one at each side. Further, they analyzed the delay budget for a series of operations and concluded that adding a Performance Enhanced Proxy (PEP) could improve the overall performance. PEPs will be introduced later in this chapter.

Another solution that could help the PS operations is to efficiently send messages to a large group of users. [31] proposes a satellite assisted push-to-send solution that helps to send messages to heteroge-

RESTRICTED DOCUMENT

neous receivers.

The cited alternatives have proven to be technically feasible. The adoption of the IP systems helps to extend the use of satellite systems. Once it is possible to convert PMR or other signals into IP packets, it is possible to transfer them through satellite networks. Nevertheless, there exist serious doubts that satellite can deliver the strict performance requirements listed by the PS organizations. The long propagation delay increases the difficulty to match the desired access times and mouth-to-ear delays. The spectrum of the satellite systems is also limited and there no exist reserved bands for PS services within the most common used parts of the spectrum. Additionally, the transmission at high frequencies is complex given the attenuation characteristics of some bands. Further, the deployable satellite terminals are exposed to a potential degradation due to antenna pointing errors. Finally, algorithms to reduce jitter and latency are necessary to improve the quality of service of voice and video services.

Therefore, there is a lack of optimization of the intrinsic functions needed for the PS operations, such as group calls, in hybrid networks (terrestrial-satellite) or mobile satellite systems. This thesis provides solutions to some of these aspects.

2.2 Professional Mobile Radio

Professional Mobile Radio (PMR) are radio communication systems basically used by first responders and other professional users. They are also known as private mobile radio or land mobile radio systems. Instead of using traditional one-to-one calls, such systems focus on point to multipoint or group communications. Generally, the push-to-talk (PTT) mechanism is used to assure that only one person is speaking in a closed group at the same time.

First systems were analog and consisted of a single base station and some user terminals. Such solutions are still used for some businesses and other consumers, which use unlicensed bands. PMR446 (because it exploits the 446 MHz band) is the code name for this frequency range and the typical used equipment is known as walkie-talkies. Analog PMR uses frequency modulation (FM).

These systems have been replaced by new digital wireless solutions that we review next.

2.2.1 Digital Mobile Radio (DMR)

Digital Mobile Radio is a recent open standard developed in the ETSI [32] that aims to replace the previous analog solutions. DMR standard development was broadly divided in three parts or tiers. The Tier 1 was designed as a low cost system, using the 446 MHz band and targeting the consumer market. Then, Tier 2 operates in a variety of licensed frequency bands, from 66 to 960 MHz. It is focused on the professional market and offers a peer-to-peer communication mode and a repeater mode as well. Finally, Tier 3 covers trunked operation in licensed bands and also supports packet data services. The use of the spectrum is optimized and it provides the means to authenticate the users and manage the calls.

RESTRICTED DOCUMENT

2.2.2 Terrestrial Trunked Radio (TETRA)

Terrestrial Trunked Radio is a standard developed by the ETSI [33] to meet the needs of PS organizations, military, government and other sensible professional organizations, such as transportation, utilities or energy industries. It is an interoperable technology and equipment from multiple vendors is available. The first generation of TETRA was deployed in 1997 and while its primary market is in Europe, many countries around the world have adopted also this technology.

It has a scalable architecture: a network can be composed by a single station or have a nation-wide coverage. It can handle big groups, up to 200 users, which can belong to multiple groups. It is interoperable with the existing public telephony networks. TETRA also features strong security for authentication, authorization and confidentiality, including voice encryption. Priority configurations are also possible when needed to allow emergency calls or access priority. The main services provided are: individual and group calls, broadcast calls, short data services like messages, and packet data exchange. However, data rates are very limited ranging from 2.4 kbits/s to 28.8 kbits/s.

It uses a frequency duplexing scheme with carriers of 25 kHz. Then, the channel is accessed through a TDMA mechanism. The modulation used is DQPSK. The frequency bands used are within the 380-960 MHz range.

A key characteristic is the direct mode operation (DMO) that allows an infrastructure-free operation. User terminals can communicate directly or through a relay or repeater. Additionally, a relay node allows an isolated terminal to connect to the network.

A more recent release of the standard, known as TETRA Release 2 or TETRA Enhanced Data Service (TEDS) [34], was conceived to improve the packet service with data rates up to a few hundred of kbits/s, using modulations like 4, 16 or 64 QAM and carrier bandwidths such as 25, 50, 100 or 150 kHz. Yet, to achieve those rates a high signal to noise ratio and the use of multiple time slots are necessary, which is difficult in the operation environment of PS agencies.

2.2.3 APCO 25

APCO-25, also known as project 25 (P25), is the standard of PMR communications used by the public safety organizations in North America. It addresses the same target market as TETRA, but they are not interoperable. P25 was developed by the Association of Public-Safety Communication Officials-International (APCO) and the National Telecommunications and Information Administration (NTIA) among other American governmental organizations.

The main idea was to improve legacy analog systems and provide functionalities focused on PS needs while improving the spectral efficiency. It is an interoperable architecture to ensure the competition between multiple vendors and expand the inter-agency communication.

It supports encrypted communications and offers individual, group and broadcast calls, as well as a messaging service. It uses a FDMA access method and a QPSK-C modulation. However, the data rates are limited to 9.6 Kbps. Another limitation is that is based on a fixed network infrastructure and a base

RESTRICTED DOCUMENT

station reaches a few kilometers depending on the terrain.

2.2.4 TETRAPOL

TETRAPOL is a PMR system that was designed ad-hoc for the use of the French PS agencies [35]. More recently, other countries have adopted it as their primary system. Its name may recall the previously discussed TETRA technology, yet it has many differences. TETRAPOL uses a FDMA access scheme with one speech or control channel per 12.5 KHz carrier and a base station can handle up to 24 radio channels. The achievable rate is 8 Kbps using a binary Gaussian Minimum Shift Keying (GMSK) modulation. As APCO 25, TETRAPOL is based on a fixed network infrastructure, limiting its operation during natural disasters. The services offered are similar to the reviewed standards: group and broadcast calls, messaging and others.

2.3 Long Term Evolution

Long Term Evolution (LTE) is the 4th Generation standard for cellular communications proposed by the 3GPP partnership. The adoption of the technical improvements developed during the first decade of the 21th century allowed to boost the exchange speed and improve the overall user experience.

During that decade the mobile data usage increased exponentially. Voice ceased to be the predominant mean of information exchange. Therefore, Internet Protocol (IP) became the protocol of choice thanks to its capacity to transport all types of traffic. Additionally, the possibility to transport voice over IP enabled a seamless integration with the rest of multimedia services.

LTE aims to deliver a downlink peak rate of 100 Mbps and an uplink speed of up to 50 Mbps. LTE exploits a much higher spectral efficiency compared to legacy systems and enlarges both transmission and reception bandwidth up to 20 Mhz, providing a number of intermediate possibilities for spectrum flexibility. Furthermore, it reduces the transition from dormant to active state up to less than 100 ms. In addition user plane latency can be less than 5 ms.

This section provides an introduction to the novelties proposed in the standard reviewing the general architecture, the physical layer improvements, the treatment of the voice services and the evolution of the multicast / broadcast capabilities. Finally, we briefly discuss about the use of LTE in conjunction with satellite links.

2.3.1 Architecture and Protocols

LTE introduces a simpler new architecture that allows reducing the operational cost of the network. In addition, it enables the shared use of a base station by multiple operators reducing the deployment costs. Further, the core network can support legacy access networks (GERAN or UTRAN) but also other non-3GPP standardized systems (such as WiMAX or cdma2000). Figure 2.1 shows the general architecture of a LTE network. Next we concisely review the job of the different entities.

RESTRICTED DOCUMENT

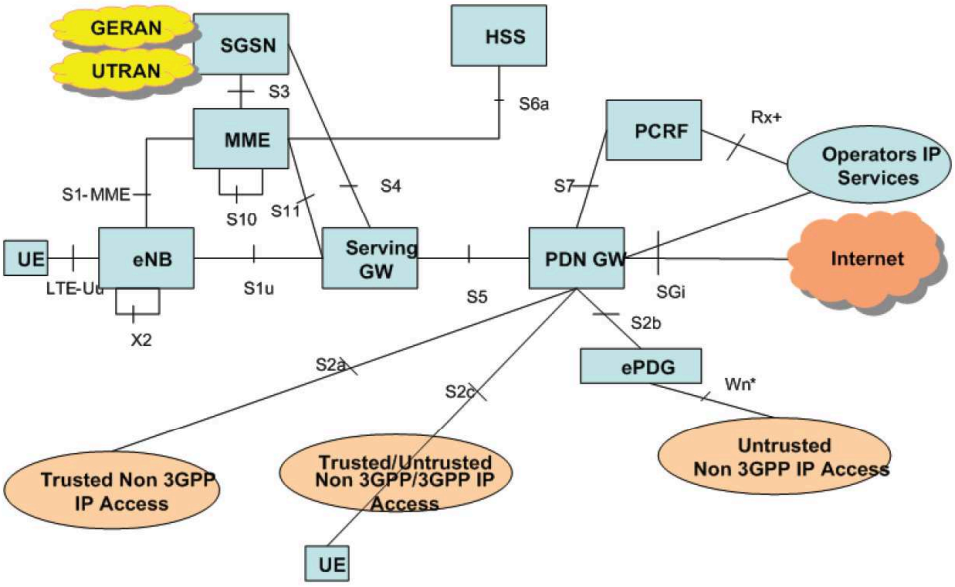


Figure 2.1: LTE Architecture [1]

The access network, known as Evolved Universal Terrestrial Radio Access Network (E-UTRAN), is composed by the user equipment (UEs) and the radio base stations, the evolved nodeBs (eNB). eNBs are in charge of the radio resource control function including the link adaptation, power control, ciphering of the data exchanged, the prioritization of the logical channels, the scheduling mechanisms and the HARQ error correction.

The main components of the core network, named as System Architecture Evolution (SAE), are the Mobility Management Entity, the Serving Gateway and the Packet Data Networks Gateway. Additionally, the Home Subscriber Server contains the central database with the user and subscription related information. The Policy and Charging Rules Function manages the policy applied to each service flow.

The MME is the control node for the access network. It deals with the tracking and paging of UEs in idle mode, the bearer activation and deactivation procedure, the choice of the SGW, the temporary identities of the UEs and the interaction with the HSS in order to authenticate the user. Moreover, it handles the intra-LTE handover and the security key management. It is also the mobility anchor for 2G/3G access networks.

The SGW is in charge of maintaining the data paths between the eNBs and the PDN Gateways, especially during user plane mobility.

Finally, the PDN Gateway provides connectivity towards external packet data networks and it is the entry and exit point of the user data. A UE may maintain a relationship with multiple PDN Gateways.

Figure 2.2 shows the protocol stack for the user and control planes. The Non-Access Stratum (NAS) protocols manage the session and the IP connectivity between the UE and the PDN GW. They control the user mobility in order to maintain an always-present connectivity. The PDCP (packet data convergence protocol) handles the header compression and security of the radio interface. The RLC (radio link

RESTRICTED DOCUMENT

control) ensure that there are no losses during the transmission of data. The scheduling and the HARQ signaling are controlled by the MAC (media access control) protocol. Finally, in the control plane, the RRC (radio resource control) deals with the mobility of users in active mode, the setup of radio bearers and the broadcast of system information.

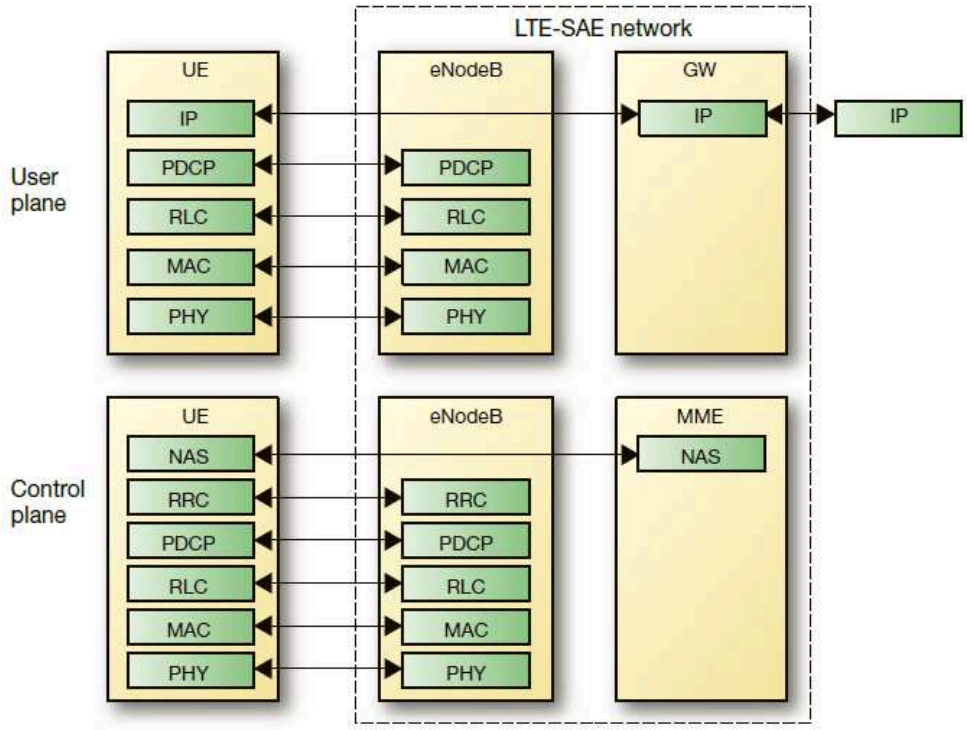


Figure 2.2: LTE Protocol Stack [2]

2.3.2 Physical Layer improvements

LTE has adopted new transmission techniques that exploit the use of multiple carriers to overcome the effects of the multi-path issues encountered in the mobile transmission channel. Legacy systems used the traditional single carrier system. These schemes are severely affected by inter symbol interference (ISI) and frequency selective fading, meaning that the bandwidth of a single carrier may have different interferences and attenuation depending on the frequency, which complicates the use of channel equalizers as data rate increases.

LTE uses two different technologies for uplink and downlink, considering the features and limitations of each transmitting entity, the eNB or the UE.

Downlink technology: OFDM and OFDMA

Orthogonal frequency division multiplexing is a transmission scheme that divides the available bandwidth into multiple carriers. The use of orthogonal subcarriers allows placing them very close one to

RESTRICTED DOCUMENT

another and achieving data rates similar to the single carrier mechanisms. Each subcarrier is modulated with QAM symbols (QPSK, 16 QAM and 64 QAM).

In the time domain, OFDM introduces a guard interval between symbols. A cyclic prefix, a copy of the end of the previous symbol, is transmitted during this guard interval. The receiver can then remove the interferences from previous symbols and improve the resistance to multi-path delays.

Considering the use of time slots and subcarriers, the available resources can be divided into resource blocks as shown in figure 2.3.

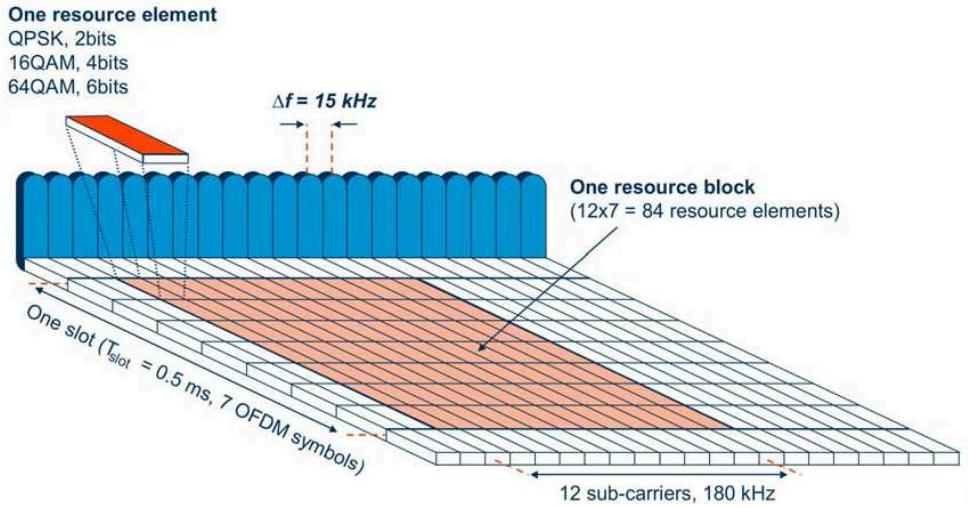


Figure 2.3: LTE Downlink Physical resources [3]

In order to serve multiple users, LTE uses a multiplexing scheme based on OFDM, OFDMA (orthogonal frequency division multiple access). This access technique dynamically assigns subsets of the subcarriers to the different users increasing the robustness of the system. It has the ability to allocate each user by frequency and avoid frequency-selective fading. Figure 2.4 displays the allocation differences between OFDM and OFDMA.

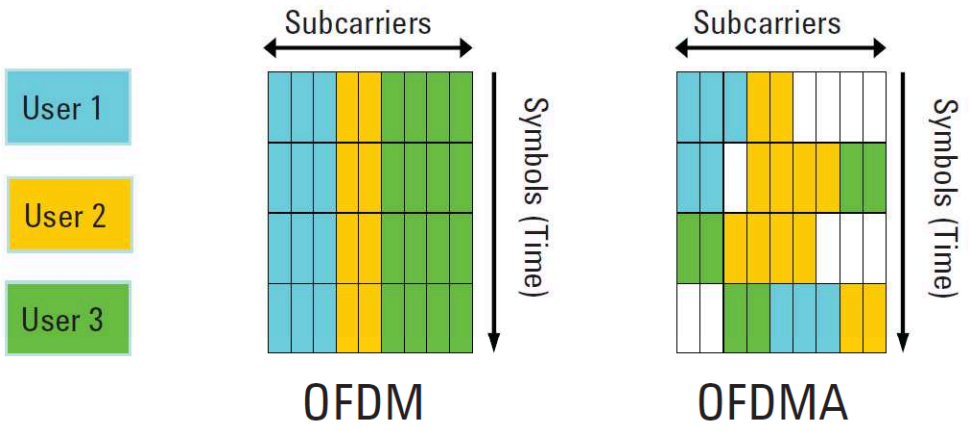


Figure 2.4: OFDM and OFDMA subcarrier allocation [4]

RESTRICTED DOCUMENT

Uplink technology: SC-FDMA

While OFDM has demonstrated its efficacy and improved throughput, it has some inconvenients that reduce its effectiveness in the uplink channel. The OFDM scheme has a high Peak to Average Power Ratio (PAPR), which increases the requirements in terms of linearity at the power amplifiers. These amplifiers are then expensive and have very high-energy consumption, which would increase the cost of user terminals and the battery usage. Instead, LTE has considered an alternative for the uplink that is based on a linearly pre-coded OFDMA and is called Single Carrier Frequency Division Multiple Access. In SC-FDMA, each user is assigned a group of adjacent resource blocks that allows the use of more power-efficient amplifiers. Figure 2.5 shows the differences between the two schemes.

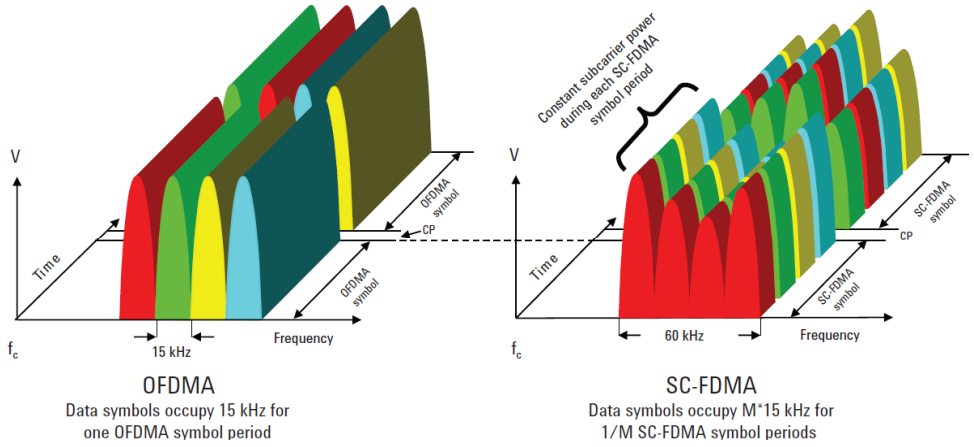


Figure 2.5: Comparison between SC-FDMA and OFDMA [4]

Multiple antenna techniques

Another key concept that increases the coverage and the overall physical layer capacity is the possibility to use multiple antennas at both transmission and reception. The multiple-input multiple-output (MIMO) techniques exploit the use of spatial multiplexing or beamforming. Multiple options are possible, depending on the number of antennas used by the transmitter and the receiver. LTE admits up to a 4x4 MIMO scheme. Figure 2.6 provides a simple overview for the combinations up to two antennas per entity.

2.3.3 Voice over LTE

Voice over LTE, or VoLTE, is the name that receives the voice services on LTE based on the IP Multimedia Subsystem (IMS). LTE foresees other methods in order to assure backward compatibility with legacy networks that employ circuit-switched mechanisms. However, VoLTE is the service that provides specific profiles for control and data planes on LTE and it was defined by GSMA in PRD IR.92 [36]. In a way, VoLTE is a form of VoIP, which will be reviewed more in detail later in this chapter.

RESTRICTED DOCUMENT

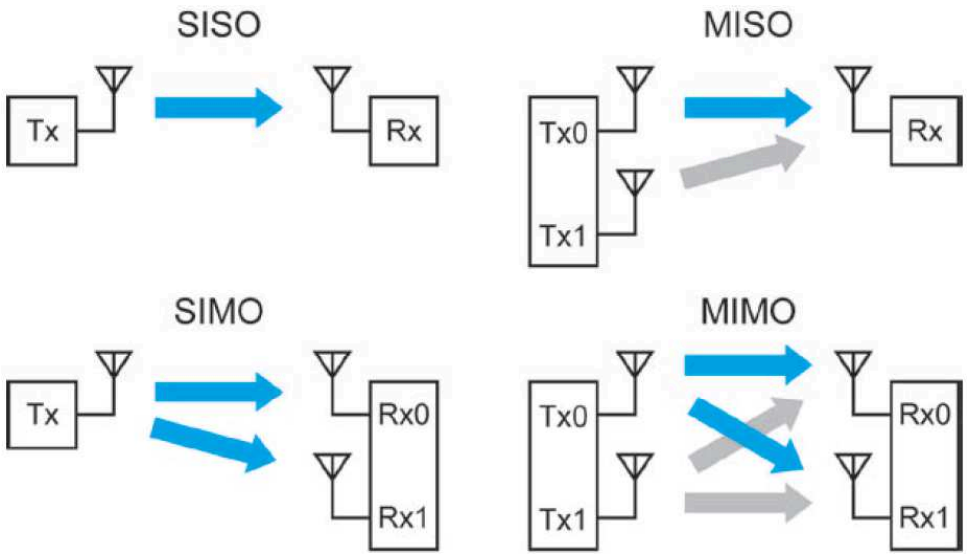


Figure 2.6: Multiple antenna combination modes [4]

Voice services can be provided as an over-the-top (OTT) application, but the integration within the IMS allows assuring a better QoS and the possibility to switch back to a legacy system compatible solution in case of handover. QoS-enabled VoLTE maintains the quality of the calls even in high load scenarios.

The main feature of VoLTE is the use of wideband encoding methods that provide a richer voice quality and are usually marketed as High Definition (HD) voice. The recommended speech codec family is Adaptive Multi-Rate Wideband (AMR-WB). Legacy codecs supported only up to 3.5 kHz whilst the wideband codecs encode until 7 kHz, improving the overall quality while maintaining an acceptable bit rate requirement.

Apart from high quality voice, VoLTE brings other interesting benefits. The call setup has been extremely reduced to less than one second, compared to up to four seconds in circuit-switched systems. Additionally, thanks to the new scheduling techniques and the discontinuous reception functionally, the terminal may switch to sleep mode between transmission and reception, improving the battery life. Figure 2.7 gives an overview of this feature.

2.3.4 eMBMS

Evolved Multimedia Broadcast Multicast Service (eMBMS) is a technology that allows LTE providing an efficient broadcast delivery service to a large number of users. When multiple users are interested in a content, the network shifts from the traditional unicast transmission towards a multicast or broadcast one. eMBMS reuses the existing LTE spectrum and enables a better performance by exploiting the flexibility between unicast and broadcast. The main target application is adaptive streaming over HTTP, but other services are in mind with special focus to video and audio distribution. eMBMS requires no changes in the consumer devices nor in the access network hardware.

RESTRICTED DOCUMENT

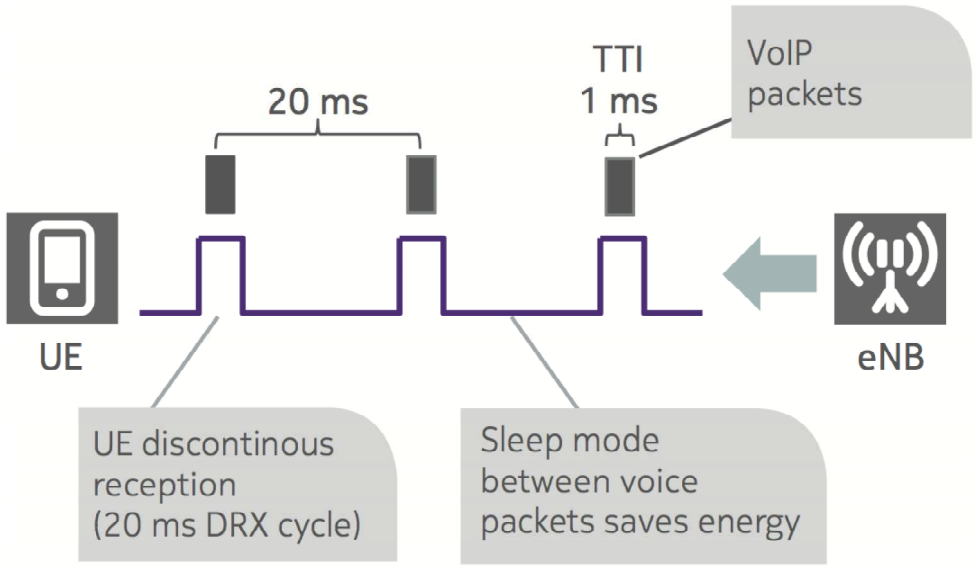


Figure 2.7: Overview of the VoLTE transmission with discontinuous reception [5]

Furthermore, the throughput is improved thanks to the adoption of the single frequency network mechanism. Multiple eNBs transmit the same content simultaneously and users can use the multiple signals to their advantage. Hence, OFDM signals are enhanced through signal combination and better throughput is achieved. The eMBMS defines service areas in which the same content can be transmitted synchronously by a number of eNB within. Overlap between single frequency network areas is possible and an eNB can belong to up to 8 areas.

eMBMS and unicast transmissions are mixed in a common LTE carrier. Up to 60% of the subframes can be allocated to the multicast/broadcast service. eMBMS reuses most of the existing architecture and adds a few entities more. Figure 2.8 presents the eMBMS system architecture.

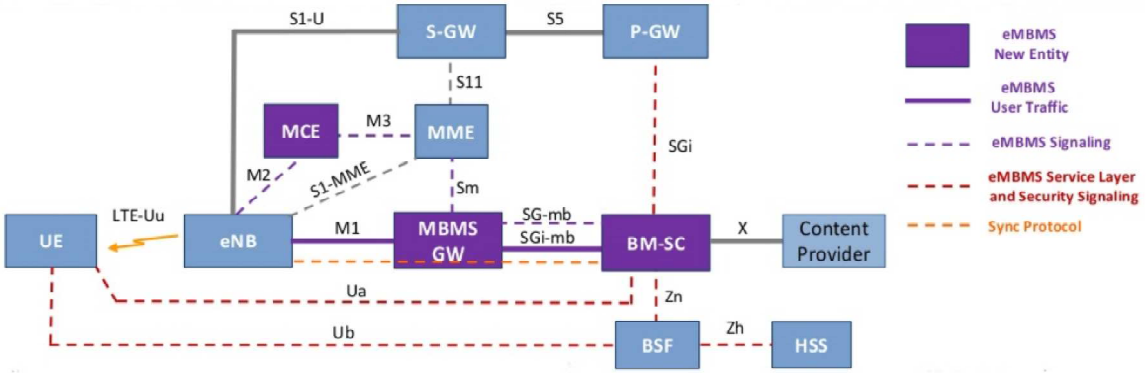


Figure 2.8: eMBMS system architecture [6]

The Broadcast Multicast Service Center (BM-SC) coordinates session setup, membership, service announcement, security and content synchronization. The eMBMS Gateway (MBMS GW) distributes

RESTRICTED DOCUMENT

the user data to the eNBs through multicast and performs session control between the MME and the eNBs. Finally, the Multi-cell/Multicast Coordination Entity (MCE), which can be part of the eNB or a separate entity, allocates the time/frequency resources for the different sessions.

eMBMS was defined in LTE Rel-9 and continues to evolve. Additional options have been introduced to provide better flexibility, avoiding to use eMBMS when there are only a few users interested in a content, or to enable priority between sessions and improve security. More recently, support for service continuity has been added to tackle user mobility throughout the network.

2.3.5 LTE over Satellite

Satellite communications enable global coverage. With the growing interest in mobile communications and LTE in general, satellite could be useful to serve users located in rural areas or in villages with small population density. Mainly, two approaches have been identified to provide a LTE-like service through satellite. In the first one, the satellite acts like an eNB and transmits the signals directly to the users; the terrestrial gateway takes care of the control procedures. Alternatively, some eNBs equipped with satellite terminals can be deployed. Then, the users communicate with regular eNB but the backhauling part features a satellite link.

Transmitting OFDM signals from a satellite is challenging because of the large Peak to Average Power Ratio. Satellite amplifiers are not designed to process such signals efficiently. However, many research studies have been devoted to the study of the performance of such transmissions. In [37], authors analyze the degradation suffered by OFDM when going through a High Performance Amplifier (HPA) such as the ones used onboard of satellites. Their simulation results show that a high peak signal creates out-of-band energy (spectral regrowth) and in-band distortion (constellation scattering) when processed by a TWTA. The signals suffered from amplitude reduction and phase rotation, penalizing the BER performance in higher order modulations. However, these effects can be reduced by introducing pre-distortion or equalizing the signal priorly. [38] studies an adaptive pre-distorter for OFDM signals for a satellite downlink channel, their analysis shows that there is still a degradation and that further work should be done. Other papers focus on the OFDM synchronization issues [39] [40]. Finally, [41] examines the performance of the LTE return channel over satellite. They experiment with SC-FDMA signals, which require frequency and timing offset adjustment techniques to adapt to the satellite environment.

An American company called LightSquared [42] wanted to provide a LTE satellite service using the empty frequency blocks near the Global Positioning System (GPS) in L-band. The venture was questioned by the spectrum coordination authorities due to the interferences it would cause and the company had in the end to fill for bankruptcy.

In LTE backhauling, the signals transmitted through the satellite link follow current standards like DVB-S2. Here the challenge is to improve the satellite part in order not to degrade the performance of the LTE part. For example, studies in [43] [44] focus on the optimization of the handover procedures. Generally, the backhauling is performed through a geostationary link but authors in [45] analyzed the

RESTRICTED DOCUMENT

performance over medium earth orbit (MEO) satellites. They targeted the deployment of multiple eNBs aboard transportation vehicles such as trains, ships or buses that provided a video streaming service. Generally the results were satisfying but in heavy load scenario the architecture was not able to meet the requirements.

2.4 PMR evolution towards LTE

In this section we first describe the benefits of adopting LTE as the main technology for future PMR systems. Then, we briefly present Push-to-talk over Cellular (PoC) as it was the first try to design a standardized solution that works over cellular networks. Next, we detail the different architectures that could provide a PMR solution over LTE. Finally, we illustrate the current work on standardization to adapt LTE to the PS requirements.

2.4.1 Reasons for adopting LTE

Need to boost data services for PS services

Nowadays, data services are becoming more important than voice-based services. While current PMR technologies supply a good base for voice exchange, they feature poor data capabilities. Such solutions provide a data rates not exceeding few tens of kb/s [33], which are clearly insufficient in the modern world where broadband access is inherently present.

Voice applications, push-to-talk and one-to-one calls, will still be the most required services for disaster management and daily operations. However, first responders demand an increase on data capabilities that PMR cannot provide. The applications that could benefit from this boost are video streaming, monitoring of location or biometrical data, geo-intelligence services such as high-resolution mapping, or file exchange.

LTE improves remarkably the speed and latency and is able to support real-time applications such real-time video and multimedia. Latency is reduced to a few tens of milliseconds and the data peak rates can achieve 100 Mbps for downlink and 50 Mbps [46]. Its improved spectral efficiency helps to reduce the cost per bit. Finally, as it is a full-IP solution, it provides an interoperable platform to supply a variety of services.

An opportunity to tackle the interoperability issue and exploit the synergy between commercial and private networks

The lack of interoperability has sometimes hampered multi-agency information exchange. In some countries, different agencies use different technologies leaving the regular one-to-one calls as the sole communication mean for cooperation. Similarly, issues appear in cross-border operations where responders from different countries use incompatible systems. Therefore, there is a need to come up with a standardized tool. The adoption of LTE is an opportunity to cope with this problem and build a solution based on

RESTRICTED DOCUMENT

the internet framework that works on top of the IP layer. In this sense, even if the access technology may be different (wired, wireless, cellular or satellite), interoperability could be assured.

To the date, private and commercial networks have pursued separate paths. This separation has not only been physical, as PS organizations demanded for dedicated networks, but also the technical. Professional networks have followed a slow standardization process compared to commercial cellular ones. This has impaired them to implement the latest technological features. The figure 2.9 compares the evolution of both systems:

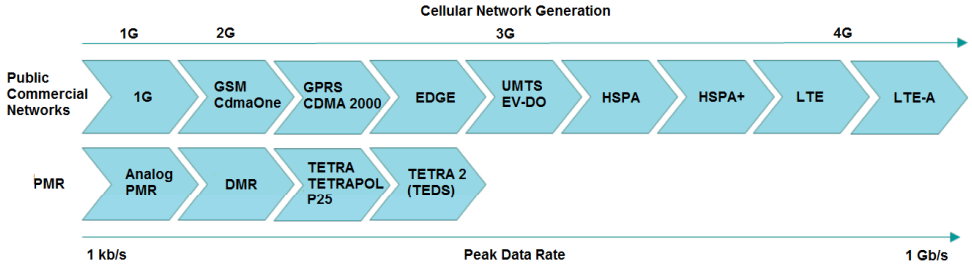


Figure 2.9: PMR and Cellular Networks Evolution. Adapted from [7]

This difference of development is, in part, due to the strong investment requirements that governments are not always able to provide and to the limited concurrency between different suppliers / operators compared to commercial systems.

The adoption of LTE as the base system for next generation PMR is a chance to create and exploit synergies between commercial operators and PS agencies. The possibility to access to the deployed public networks and more accessible equipment could reduce drastically the investment and operational costs. Furthermore, organizations could still utilize dedicated networks where they feel there is a strong need. Additionally, first responders will profit from the aggregated capacity, enhanced coverage and improved resiliency. Finally, the PS community could benefit from the faster innovation and competition of the open mass market and reduce their dependence from the niche and customized products. Authors in [47] review the principal advantages of the PMR and LTE merge from the techno-economic perspective.

Is LTE ready to match PMR capabilities?

LTE has been designed and optimized for commercial usage, so it is necessary to study the adaptation needed to fulfill the PMR constraints. Currently, LTE lacks many of the particular capabilities required for mission critical users. There is no mean to provide reliable and secure push-to-talk for group communications with a fast call set-up. Another strong prerequisite is to improve the infrastructure resilience and availability. Further, there is a need to provide an efficient point-to-multipoint communication and group management capabilities. Finally, the possibility to connect to user terminals (direct-mode) is another key requirement. A report from the National Public Safety Telecommunications Council (NPSTC) gives an extensive overview of the requirements of PTT over LTE [48].

RESTRICTED DOCUMENT

Critical voice services over LTE will be limited in early implementations and full PMR functionality will take long to replicate. At the beginning, LTE is expected to complement but not replace existing legacy PMR networks. Interworking between systems will be mandatory during this phase. This means that users will rely on legacy solutions for their voice needs and switch to LTE for data demand.

3GPP is currently developing some aspects within the LTE standard that will reduce the gap between PMR and LTE to provide mission critical applications. At the end of this section we review the main specifications that will allow implementing a PMR-like solution over LTE.

Nevertheless, standardization advances faster than deployment. Work within LTE Releases 12 and 13 are expected to include most of the required items by 2015-2016. Full compliant and nation-wide network deployment could not take place until 2020-2025.

2.4.2 PoC – Push to talk over Cellular

Push-To-Talk over Cellular (PoC) is an IP-based service proposed by the Open Mobile Alliance (OMA) [8]. It uses standardized protocols, such as Session Initiation Protocol (SIP) and Real-time Transport Protocol (RTP), to implement a PTT-like solution over cellular networks. Its architecture matches the idea of using a cellular network infrastructure and provides the PTT service as an overlay application. It was not specifically designed for PS purposes and it targeted preferably professional customers.

It works in conjunction with the SIP/IP core, known as the Internet Protocol Multimedia Subsystem (IMS) in 3GPP networks. Figure 2.10 presents the architecture of the service and relationship between the functional blocks.

The three PoC-specific functional elements are the PoC client, located on the user's UE, the PoC server that facilitates the communication between users, and the PoC Box that interfaces with the groups database and user policies. If necessary, multiple PoC servers can participate in a session. One of them will designated controlling PoC server and will be seen as the central manager of the session.

Several vendors commercialize solutions based on the implementation of the PoC service. It is unclear, however, that it could fulfill the prerequisites of the PS community. Feasibility and delay analysis have been conducted in this field [49] [50]. The adoption of LTE makes PoC a promising option as the analysis concluded that it is closer to satisfy the requirements.

2.4.3 Architectures to provide Public Safety communications over LTE

We have seen the benefits of having LTE as the vehicular technology for PMR systems. There exist, however, several alternative ways to address the implementation of PMR services over LTE. In this subsection, we review the different approaches that we have identified and discuss which way seems the most promising from our perspective. A wider list of references can be found in our white paper about Public Safety communications over LTE [12].

First, we give an overview of the most typical current situation: the coexistence between LTE and PMR networks. This scenario considers that PMR is used for voice communication and LTE remains an

RESTRICTED DOCUMENT

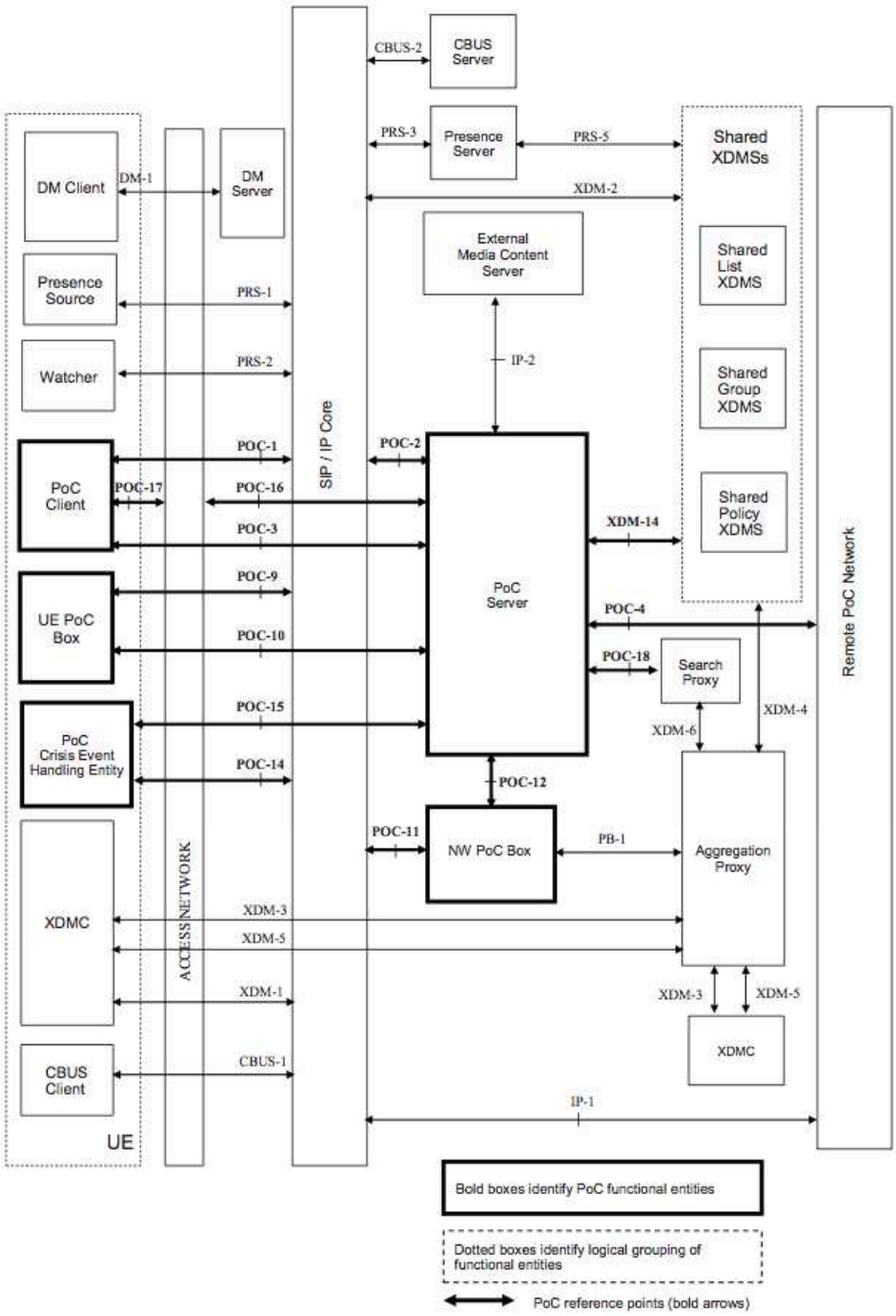


Figure 2.10: PoC Architecture [8]

option for faster data exchange. The two networks are separated with no real interconnection. Reserved spectrum bands for LTE could be used if available and dedicated cells (femto/satellite) would allow reconnecting isolated areas. Yet, this is not a real solution as PMR services are not provided in any way through the LTE technology.

RESTRICTED DOCUMENT

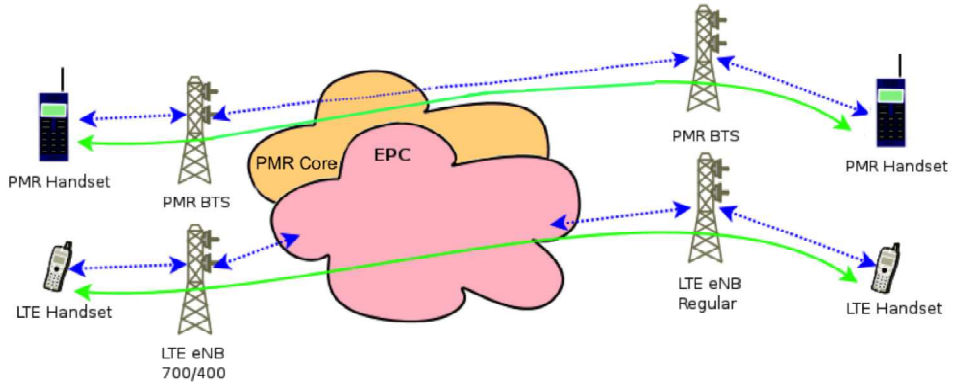


Figure 2.11: Overview of the current PMR and LTE coexistence

Interconnection between a PMR access network and a LTE core

The first alternative is a transitional solution that makes possible to still use and amortize legacy investments while starting the convergence between PMR and PLMN network cores through LTE. Responders use regular PMR handsets and PMR BTS interconnect with an LTE EPC.

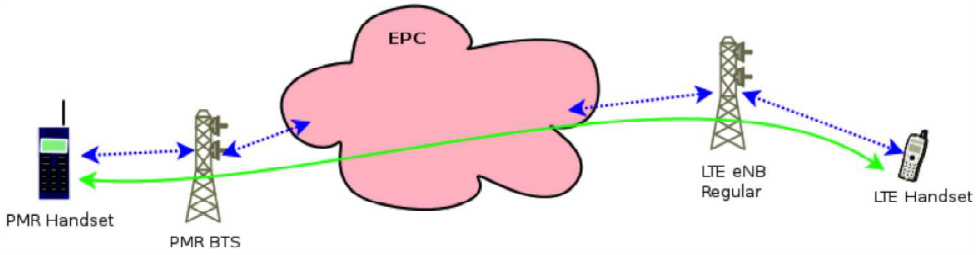


Figure 2.12: PMR Access Network with a LTE Core

This approach is interesting because it does not obsolete the installed equipment and copes well with an incremental rollout of LTE networks. Still, it does not solve the data-bandwidth bottleneck at the PMR level. Additionally, QoS compatibility needs to be ensured between LTE and PMR (e.g., delays for group call setup).

Several systems have been implemented under the paradigm of Radio over Internet Protocol (RoIP), which defines the concept of VoIP with PTT. It is based on switches that are connected to PMR base stations and translate PTT messages to valid IP packets, so they can be transported through IP networks.

Regular LTE with PMR as an overlay service

In this case, PMR service is provided as an over-the-top (OTT) application. LTE is used as the access and core solution and PMR is implemented on the application layer. In case of network collapse, since the PMR overlay is IP based, another bearer such as WLAN can be used in conjunction with satellite backhauling of IP traffic.

The actual implementation of the overlay PMR service is not standardized. Several third-party applications provide a PTT-like service but it is unclear that they fulfill the actual PMR requirements. Another possibility is to use PoC.

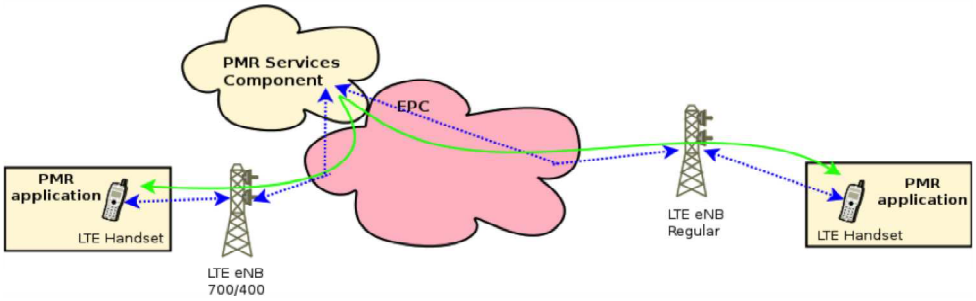


Figure 2.13: Architecture of LTE with PMR as an overlay service

It is an interoperable solution based on off-the-shelf equipment where one device fits all usages. However, direct mode is still not supported and, similar to previous case, it raises concerns about the QoS performance.

As it has been reviewed earlier, several vendors propose solutions based on PoC. But other third-party applications are more focused on the mass market and do not target PS customers specifically.

PMR-enabled LTE

This last option is a "from scratch" stance. Future LTE standards and deployments will feature PMR services directly integrated as LTE core applications. Responders will use LTE handsets with PMR extension and these will provide support for group calls and direct mode. As before, one device will access all necessary services. Most of the requirements will be satisfied via the optimization of the LTE network.

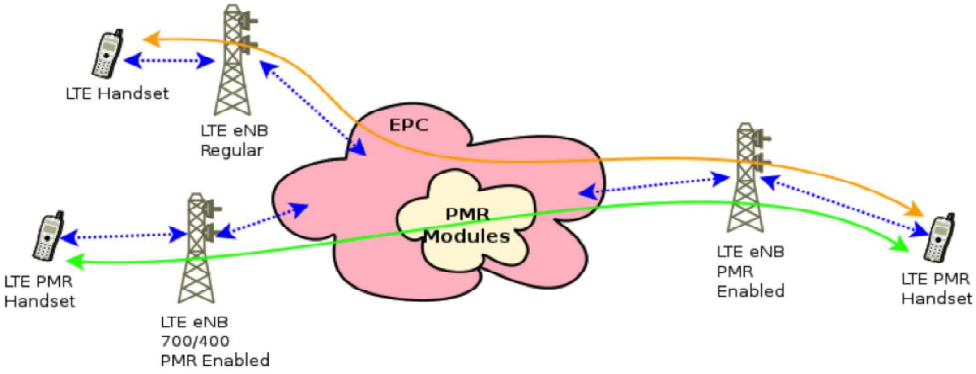


Figure 2.14: Architecture of PMR-enabled LTE

This integration looks appealing from a performance standpoint but it might obsolete recently deployed LTE equipment and legacy PMR equipment. In addition, there is still a significant amount of

RESTRICTED DOCUMENT

work to address all the requirements within the standard. Next subsection reviews the progress on this subject and the different working groups involved.

It is the final stage but the process will take time. Major investments on legacy PMR networks are still ongoing and the standard will not be ready at least until the next release, R13, which is scheduled for 2016. We do not expect important deployments until 2020-2025.

2.4.4 3GPP work to integrate PMR within LTE Advanced

3GPP, as the standardization body for LTE, is improving its specifications and solutions to finally integrate PMR as a core application. The first efforts were conducted during release 12, planned for 2014, but it is expected that full compliance will arrive with release 13 in 2016. The different actions can be divided into two subgroups: group communications or push-to-talk and improvements to network resiliency, such as the direct mode. The concerned working groups and codes are:

- Group Communications Service Enabler over LTE (GCSE-LTE) – TR 23.768 [51]
- Mission Critical PTT for LTE (MCPTT) – TS 22.179 [17]
- Proximity Services (ProSe) – TR 22.803 [52]
- Isolated E-UTRAN Operation for PS (IOPS) – TR 22.897 [53]

Next, we provide an overview of the main subjects studied by these groups.

Group Communications Service Enabler over LTE

This group studies how to efficiently provide group communications over LTE. It describes a high level reference architecture that interfaces with the LTE EPC. The two main elements are the GCSE Application Server and the Multipoint Service (MuSe). The first one will carry the actions concerning group management, media handling and floor control and the latter will be in charge of interfacing with the multicast function benefiting from the eMBMS implementation. Additionally, a GCSE application will be located in the UE. The uplink towards the server is expected to be a unicast link while the downlink to users could be unicast or multicast.

The key issue here is to decide when to use unicast or multicast depending on the scenario and the corresponding interaction with the eMBMS service. Other elements relate to the group management, privacy and security, as well as prioritization. Furthermore, it addresses the problem of service continuity through proximity services or roaming.

Mission Critical PTT for LTE

This group analyzes the requirements to provide a push-to-talk service over LTE. It leverages the work done in GCSE and ProSe and stipulates some specific requirements for the PTT application. It deals with group management, priority, the different types of calls and floor control.

RESTRICTED DOCUMENT

Proximity Services

This group studies the feasibility of a direct communications mode over LTE. It includes in addition the possibility to connect with nearby UEs in a locally routed manner, i.e. involving only the eNB without interacting with the core network.

The document [52] analyses several use case for both general and PS users. The technical specifications for direct communications are still on the research field. It is considered that a strict direct mode will only be enabled for PS users, inside or outside of network coverage. Another key element is the discovery of other UEs nearby, which could be network assisted or strictly direct. Then, in the first case and considering regular users, the network shall authorize the direct communication between devices.

Lastly, the service requirements for the Evolved Packet System (EPS) regarding the coordination with nearby eNBs are discussed in TS 22.278 [54].

Isolated E-UTRAN Operation for PS

The document investigates the situations that could be improved by locally routed communications for public safety UEs. An isolated E-UTRAN is an access network that is not able to connect normally with the core. It can include one or multiple eNBs.

This group examines the possible use cases and the requirements to enable the different solutions for the isolated areas and improve the overall resiliency of the network.

2.5 Voice over IP

Voice over Internet Protocol (VoIP) is the set of mechanisms and protocols that provide voice communications over the Internet. First technologies for voice communications were based on circuit-switched networks as the public switched telephone networks. In such networks, during a call, the users reserved some of the resources to establish the communication. These resources were dedicated to the given call until it finished. Hence, no other users could use them leading to an inefficient use of the uplink/downlink resources.

Internet telephony was born to overcome these issues and transport voice communications via the internet, which is a packet-switched network. VoIP not only refers to internet telephony but to the carriage of voice signals over IP networks. Voice is digitalized, encoded and finally encapsulated in packets that are transmitted over the internet. This allows different users to share the network resources by intercalating packets coming from different sources over the same band.

2.5.1 Protocol Stack

Figure 2.15 shows a high level view of the VoIP service protocol stack.

In the application layer point of view, VoIP systems employ session control protocols to control the set-up and tear-down of calls as well as audio codecs which encode/decode speech. The choice of

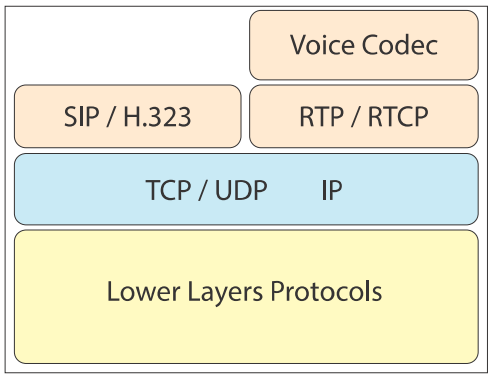


Figure 2.15: VoIP protocol stack

codec depends on the application requirements and the network resources. It is possible to have different choices within the same application in order to adapt to the network. We review the most important codecs in the next subsection.

Voice communication is time-sensitive rather than content-sensitive and, therefore, it is classified as real-time. Real-time traffic can tolerate some packet error but it is delicate concerning delay and jitter. Voice is encapsulated in a standardized packet format called Real-time Transport Protocol (RTP). RTP is used conjunctly with the RTP Control Protocol (RTCP), which deals with the monitoring of the quality of service (QoS) and the synchronization of multiple media streams. Once the session is established, the voice data that is carried by RTP is exchanged directly between the different parties involved in the call.

The protocol that generally deals with the session set-up, management and tear-down is the Session Initiation Protocol (SIP). The session can be between two parties (unicast) or multi-party (multicast). SIP is independent of the underlying transport protocol and it can run on Transmission Control Protocol (TCP), User Datagram Protocol (UDP) or Stream Control Transmission Protocol (SCTP).

SIP architecture is defined by the following elements: User Agents (UA), registrar, location, proxy and redirect servers. The user agent is the network end-point that creates and receives SIP messages. The registrar deals with the registration and update of users and services within a database. Location server tracks the users' location. The proxy server forwards the messages between UAs and the Redirect server forwards the requests to other proxy servers if the destination address is not in the database.

An alternative to SIP is H.323, a recommendation from the ITU Telecommunication Standardization Sector (ITU-T). It defines the protocols to provide audio-visual communication session control on packet networks, which can also be either point-to-point or multi-point conferences.

Regarding the transport of the voice packets in the RTP format can also use TCP and UDP. However, TCP is not usually used in RTP applications because TCP favors reliability over timeliness. Generally the RTP implementations are built on the UDP.

RESTRICTED DOCUMENT

2.5.2 Voice Codecs

Voice codecs is the name that commonly receive the algorithms that are coding or decoding a digital data stream of audio. In this way, audio information can be compressed and packetized. An audio codec always searches a tradeoff between the quality of the audio signal and the number of bits of information per second, the bit rate, that is needed in the application layer. The general rule of thumb indicates that the higher the bit rate, the better quality the codec delivers. However, there are some exceptions depending on the coding principle.

Codecs can be classified between the ones that only have one available bit rate, constant bit rate (CBR), and the ones whose bit rate can change or adapt depending on the situation or congestion, variable bit rate (VBR). The codecs used for telephony or audio communication can also be sorted according to the audio bandwidth they process. On one hand, we have narrowband codecs with a passband of $300 - 3400$ Hz. They have been the common target in legacy voice services. On the other hand, more recently, wideband codecs have enabled the possibility to process a higher band, from 50 to 7000 Hz, and reach a superior quality. Tests have shown a noticeable difference in quality of experience comparing to narrowband codecs with a similar bit rate.

The use of these codec algorithms introduces an additional delay due to (de)codec buffering and processing. Codecs are also defined by their frame size, which indicates the quantity of audio time that is included in a frame. The typical values are $20 - 30$ ms per frame. Another point to consider is the computation complexity. Some of the codecs reach very low bit rates at the cost of high processing complexity. This could be an issue in low capabilities devices.

Some of the coding formats are proprietary and others are fully standardized. These last ones are usually created by the ITU-T and receive a code name under the G-series [55].

Adaptive Multirate (AMR) was the first codec created by the 3GPP for 3G mobile services. It is a narrowband codec with bit rates ranging from 4.75 to 12.20 kbps and 20 ms frames. It offers a good compromise between complexity and quality. It exists a wideband codec (AMR-WB) that is featured in LTE and improves the performance with bit rates between 6.60 and 23.85 kbps. For GSM technology, there exist different codecs, but they are clearly inferior in terms of quality.

Considering the ITU-T Gseries, a large number of codecs are available, depending on the release date and the targeted application. In the case of telephony, G.711 is the classic lossless codec with a 64 kbps bit rate. Newer VBR codecs have been introduced and are capable to match AMR performances. G.729 family features codecs in narrowband and wideband with bit rates from 8 to 32 kbps. G.729 offers a better error tolerance compared to the legacy G.711. Finally, we find the G.718 family, a highly resilient coding standard that offers a scalable solution for compression of 8 and 16 kHz sampled audio signals with bit rates between 8 to 32 kbps, as well. G.718 is also interoperable with AMR-WB at 12.65 kbps. It was designed as a layered protocol suited for congestion control and differentiated QoS. It is possible to code at high quality with all the layers and discard the higher layers if necessary. Only the lowest layer, the core layer, is strictly necessary. [56] ranks the most common codecs by their subjective audio quality.

RESTRICTED DOCUMENT

2.6 Protocol optimization to the satellite environment

2.6.1 Satellite links issues

Most of the protocols used nowadays on internet networks were designed for terrestrial wired networks. Satellite networks, in addition of having wireless links to transmit to or receive from the satellite, present some specific characteristics that we review next:

- *Long propagation delay:* Internet over satellite is usually provided by geostationary satellites, which orbit 35.786 km above the equator following the rotation of the Earth. This allows using fixed satellite terminals. However, the long propagation delay is a major drawback. The time needed for a signal to reach a GEO satellite and be retransmitted back to the ground, known as the Round Trip Time (RTT), is approximately 250 milliseconds. Additionally, the delay of the resource access techniques may increase the overall delay considerably.

- *Asymmetric connection:* Generally the bandwidth used for uplink and downlink are not the same. Moreover, the technology used in the forward (from a gateway to a user terminal) and the return (from the user to the gateway) links does not offer the same performance. Plus, the performance of the satellite terminals is limited in order to reduce the cost.

- *Bursts errors:* Satellite links' bit error rate depend highly on the weather conditions and the frequency used. Those environmental conditions affect the final availability of the link, which differs from the one wired links offer. Typical overall bit error rate using the current technology is very low, in the order of 10^{-7}, however when errors occur in bursts, retrieving the correct frames using coding techniques becomes more challenging.

TCP Behavior over satellite

Nowadays, Transport Control Protocol (TCP) remains a widely used protocol especially in no real-time applications that are demanding ubiquitous broadband access. Yet, TCP has a poor performance over satellite links.

Next, we discuss the TCP mechanisms that are deeply limited by the different factors we presented previously. A more comprehensive explanation may be found in [57].

- *3-way handshake:* TCP connection procedure needs to be done even if very short data needs to be transmitted. The connection establishment takes a few seconds over satellite links.

- *Slow Start and Congestion Avoidance:* Slow start is used to gradually increase the sending rate. At the beginning, the sender sends a single segment and exponentially increases the sending window in a RTT basis, upon reception of the corresponding ACK. If an error is detected before the window achieves a certain threshold, the process restarts. Once that threshold is reached, congestion

RESTRICTED DOCUMENT

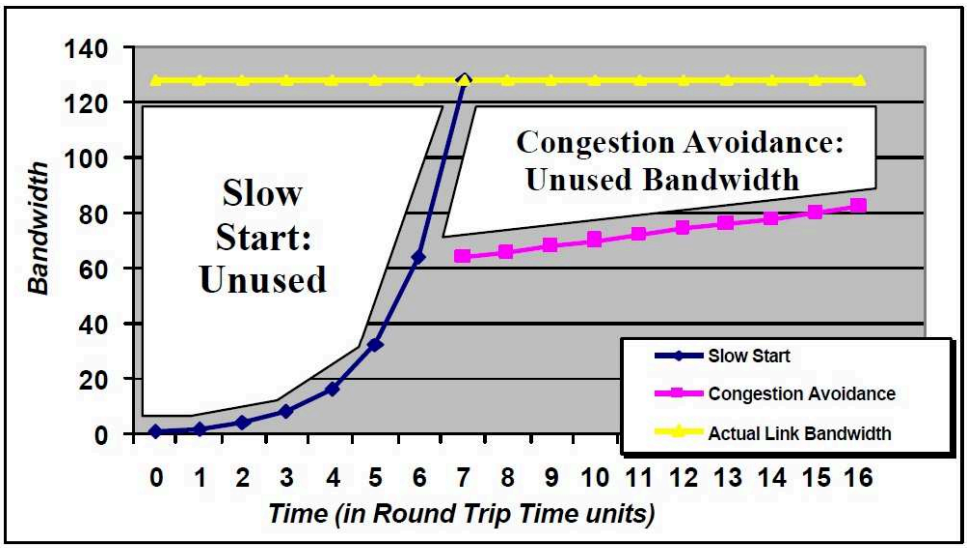


Figure 2.16: Overview of the TCP slow start and congestion avoidance techniques

avoidance mechanism takes over and reduces to the half the sending window in case of an event and increases it slowly after it.

The time to reach the maximum sending window on both mechanisms depends on the RTT, which is clearly high for GEO satellite links. Plus, in presence of bursts errors, TCP is not able to identify whether it comes from congestion on the network or a sporadic error. TCP always assumes congestion and forces the sender to operate below its actual capacity.

- *Window Size:* The transmitting throughput is defined by the receiver buffer and the RTT. The maximum receiver buffer size is 64 KB with a satellite link delay of 500 ms (two-way), the achievable throughput is approximately 1 Mbps. Even though it exists the option of window scale to increase the receiver buffer, it is a decision of the application to benefit from this option.

- *Retransmission Time Out (RTO):* RTO computation is based on the RTT value. A long RTO means that the sender is waiting longer to retransmit a corrupted segment. If the topology of the network changes or handover, the RTT may increase and unnecessary retransmissions may occur.

- *Asymmetric bandwidth:* A low bitrate in the return side may limit the sending capacity because of the congestion of ACK packets.

- *Cumulative ACKs:* The first implementation of cumulative ACKs is limited in the information it can provide on the total number of lost packets. Unnecessary retransmission may occur.

- *Delayed ACKs:* The receiver may delay the ACK feedback to send it along with data going on that direction. A timer defines the maximum waiting time for the data. However, this increases the final delay of the feedback and may incur additional issues.

RESTRICTED DOCUMENT

- *Variable bit-rate:* Latest satellite technologies, such as DVB-S2 or DVB-RCS, incorporate new Adaptive Modulation and Coding techniques (ACM). In case the transmission conditions improve, TCP will not be able to rapidly benefit from the increase of bitrate. On the other hand, a decrease of bandwidth would trigger the congestion avoidance mechanism.

Studies have been carried out to provide efficient solutions to the cited issues. These can be roughly classified in two groups: the ones that attempt to modify and extent the TCP standard and the ones that do not require TCP modifications and actually break the notion of an end-to-end link and try to isolate the constraining parts of the link. We next analyze these different approaches.

2.6.2 Protocol Alteration: TCP Modifications

Several new protocols based on TCP but that address some of the listed problems have been proposed. We briefly explore the main elements of the most commonly used:

- *TCP Reno:* It introduced the technical of Selective ACK (SACK) in order to efficiently inform of the lost packets. It also performs Fast Recovery in case three duplicate ACKs are received. It limits the reduction of the congestion window and retransmits the missing packets. If it correctly receives the ACK, it returns to the congestion avoidance phase and if not it enters the slow start state. The last modification, under the name of New Reno, can be found in [58], which transmits an additional packet together with the missed one.

- *TCP Cubic:* It uses a window that is a cubic function of the time since the last congestion event. The window increases rapidly until a congestion is detected, then it has a convex growth where CUBIC looks for more bandwidth, allowing the network to stabilize. Additionally, Cubic does not wait for the receipt of ACKs to increase the window size, depending only on the last congestion event.

- *TCP Vegas:* At first, TCP measured the RTT based on the last transmitted packet in the buffer; TCP Vegas measures the round-trip delay for every packet and uses additive increases in the congestion window.

- *TCP Peach:* It is composed of two algorithms named sudden start and rapid recovery, which are designed to substitute the classic slow start and fast recovery ones. It uses dummy low-priority segments to probe the availability of the link. If they are correctly acknowledged, the start of the transmission of the actual packets can be accelerated.

- *TCP Westwood:* It tries to overcome the issues of the losses in wireless channels. It introduces a modification called faster recovery that sets the slow start threshold by a function of the estimated available bandwidth limiting the slow-down derived of the sporadic channel losses.

RESTRICTED DOCUMENT

- *TCP Hybla:* This modification [59] targets the links with high-latency, like in the satellite case. It evaluates analytically the congestion window dynamics and attempts to remove the performance dependence on RTT.

- *SCTP:* The Stream Control Transmission Protocol was designed for voice over IP applications, but it has been used for a wider set of services. It retains some of the standard TCP algorithms and adds some new options regarding multi-homing and multi-streaming. With multi-homing, nodes may have multiple addresses and set multiple associations with other end-points trying to introduce a sense of space diversity. Multi-streaming assures that the delivery of each sub flow is independent of the others, alleviating the issues of strict byte-order delivery. Other security improvements were also proposed.

2.6.3 Performance Enhanced Proxy (PEP) Solutions

PEPs are network agents conceived to improve the end-to-end performance of some communication protocols. PEPs work by splitting the connection into multiple parts and optimizing the protocol considering the characteristics of the different parts. This is interesting in networks that involve different environments with diverse link characteristics (wired, wireless and satellite). PEPs operation is transparent from the end points and they do not need to know about their existence.

PEPs can work in the transport or application layer. The most typical case is the enhancement of the widely used TCP, without modifying the application in any way. However, PEPs may operate in the application layer and try to reduce the overhead introduced in such layer in order to provide some extra features that might be unnecessary and/or inefficient in some link conditions.

PEP has been an extensive subject of study and there exist a large number of references and approaches that build upon the PEP concept [60] [61] [62] [9].

PEPs have two different implementations: integrated and distributed. In integrated PEPs, their functions are implemented in a single node and therefore, they divide the end-to-end connection into two parts. Instead, in distributed PEPs the functions are implemented in multiple nodes and are generally used to surround the concerned link, i.e. the satellite link, by placing two PEP agents at each end of it.

PEP agents manage two different connections, which can use two different protocols of the layer the PEP operates on or one can use an enhanced or modified version of the standard protocol used in the other leg, like the ones we reviewed briefly. Figures 2.17 and 2.18 show the integrated and distributed architectures respectively:

PEP specific mechanisms

- *TCP Splitting:* The splitting approach is to separate the satellite link from the rest of the network seen in distributed PEPs. The PEP agents act as gateway and create separate connections between the endpoints. Two connections link the hosts and the PEP agents and another one link between the two agents through the satellite. This defines a new architecture and hides the satellite portion

RESTRICTED DOCUMENT

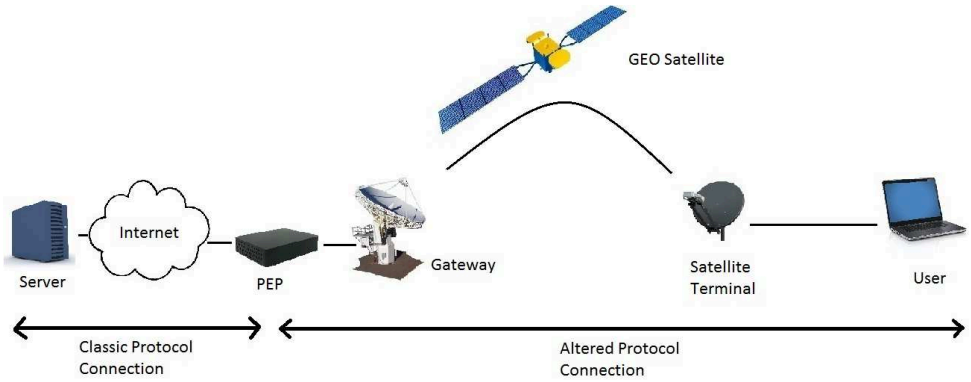


Figure 2.17: Overview of the Integrated PEP architecture

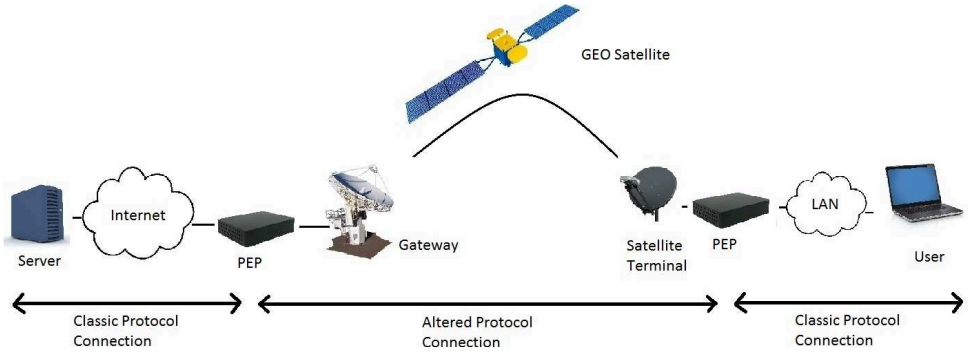


Figure 2.18: Overview of the Distributed PEP architecture

to the rest of the network. End users use standard TCP connections with the agents whereas the protocol used through the satellite is designed to overcome the intrinsic characteristics of the link.

- *TCP Spoofing:* This mechanism consists in supplanting the behavior of the end receiving part. When a PEP agent receives a data packet, before forwarding it to the actual destination, it sends an acknowledgement packet to the sender that believes the packet arrived successfully to the receiver and continues transmitting new data. The agent keeps a copy of the packet in its buffer until it receives an ACK from the receiving user or another PEP agent in the case of a distributed PEP solution. As we can see in figure 2.19, this creates the illusion of a shorter RTT and increases the amount of data transmitting over the end-to-end network.

- *Other Techniques:* Splitting and spoofing are the main mechanisms of every PEP solution. However, there are some other tools that may be added to enhance the overall protocol:

  - ACK Spacing: In environments with large bandwidth and delay, the ACKs tend to reach the source together. ACK spacing is used to separate the ACK packets and smooth the flow of these packets.
  - ACK filtering and retransmission: If the return link has a limited bandwidth compared to the

RESTRICTED DOCUMENT

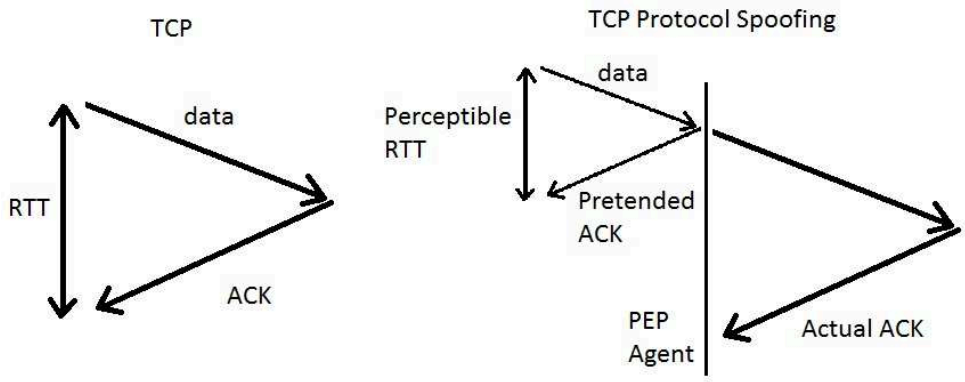


Figure 2.19: Overview of TCP Spoofing [9]

forward link, TCP ACK may get congested rapidly. This technique filters the ACK packets and transmits periodically an ACK packet that account for several regular ACKs, reducing greatly the amount of traffic dedicated to those packets.

- Error control: PEP agents may use selective acknowledge (SACK) or negative acknowledge (NACK) instead of regular ACKs. A SACK packet shows which packets have been received correctly and which have not been received. Alternatively, it may send a NACK when it detects discontinuous sequence numbers to indicate which packets were missing. Both techniques allow reducing the traffic in the return side.
- Tunneling: A PEP agent encapsulates the messages to force them go through a specific link. Another agent removes the encapsulation before forwarding to the end-node.
- Compression: With the objective of reducing the number of bits that will be transmitted to a problematic link, a PEP agent compresses the packets to improve bandwidth-limited links.
- Handling Disconnection Periods: In case the link is not available and the excepted ACKs are not received, a retransmission timeout occurs and triggers the congestion avoidance. A PEP agent may freeze the connection keeping the last ACK received, preventing the RTO to expire, and continuing the transmission later.
- Priority-based Multiplexing: When the connection is shared between different users and traffic types, the objective of this tool is to give a priority to urgent packets (interactive applications) and delay the rest.

Evaluation of PEP

Several research papers evaluate the performance and usefulness of PEPs when used together with other techniques or TCP versions [63] [64] [65] [66]. We briefly review the main conclusions they extract:

- Spoofing is beneficial in the case of large file transfers. For small transfers, it increases the throughput seen from the sender perspective. However, it seems less beneficial from receiver standpoint,

RESTRICTED DOCUMENT

which only needs to send small packets (ACKs). Spoofing creates a bottleneck in the PEP agent and may cause dropping of some packets degrading the performance perceived by the receiver.

- Splitting is a good technique to improve the throughput and reduce the latency. A split connection performance improvement is better for a large file than for a small one. Yet, it is more sensitive to congestion and its enhancement will decrease faster than with an end-to-end connection.
- Splitting improves the value of caching. If a cache hit occurs in a PEP agent, just before the satellite link, the throughput increases. If there is no splitting, a cache hit does not provide a significant improvement.
- In the case of web browsing, using HTTP, the improvement of PEP is reduced as the number of embedded objects (images, video...) increase. A different TCP connection needs to be established for each object. It is preferable to use a persistent connection, were the webpage is transmitted as a whole. This is consistent with the other conclusions that say that the improvement is higher with large files.

RESTRICTED DOCUMENT

Chapter 3

Architecture of a PTT Solution on a LTE and Satellite Hybrid Network

This chapter describes the process followed to choose an architecture for providing a push-to-talk solution over LTE and satellite. First, we review the options from the network point of view. Then, we discuss the high-level architecture explaining the entities that will be on charge of the optimizations proposed in this thesis. Finally, we compile the previous information and present the final choice.

3.1 LTE and Satellite Hybrid Network

Hybrid networks are based on two or more telecommunications technologies. The different options depend on the network architecture, the position of the satellite link and the deployment possibilities.

3.1.1 Integrated Hybrid Network

Integrated networks are characterized by the direct communications between mobile terminals and the satellite. In the case of hybrid networks, in order to be able to connect with the LTE network also, terminals are dual-mode, i.e. they can access to both technologies. For the satellite part, a mobile satellite system is used. Some of these systems are based on low earth orbit (LEO) constellations such as Globalstar or Iridium, which feature a much lower round-trip-time. Yet, we can also find geostationary satellite constellations, like Thuraya or Inmarsat, that target the mobile market. This architecture can be observed in figure 3.1.

This architecture offers a worldwide coverage exploiting at the same time the satellite and the cellular network possibilities. Zones with low density of population or of difficult access can be served by the satellite while the commercial LTE network can be used in urban environments. Another advantage is the instant deployment, given that the necessary infrastructure is already in place and it is expected that more areas will be covered by commercial cellular networks. Further, it is possible to efficiently use the best of both networks in case the user terminal has access to them. Given the broadband/multicast nature

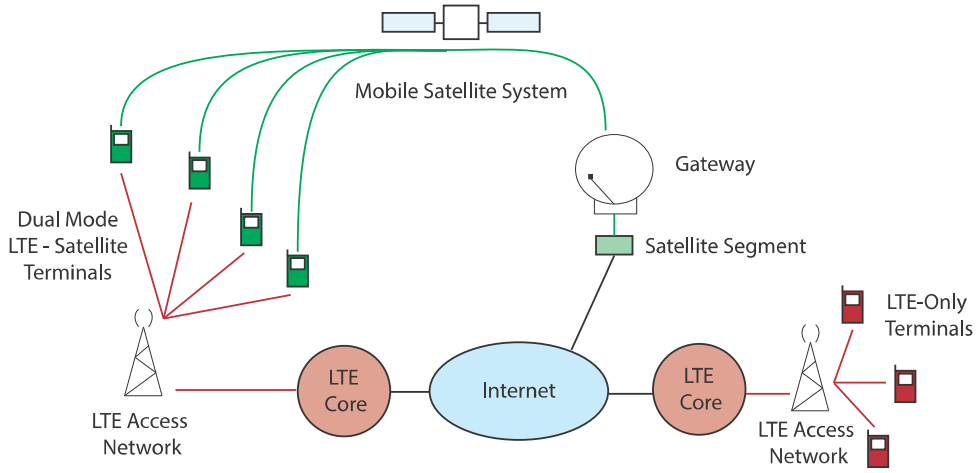


Figure 3.1: Integrated Hybrid Network

of satellite, downlink traffic can be derived through it and the uplink, from the terminals to the rest of the network, can be received by the LTE network.

While this option looks very appealing at the beginning, there exists a major issue concerning the terminal cost and availability. One of the reasons for PMR networks to evolve towards a full-IP technology like LTE was to reduce the terminal cost. A dual-mode terminal such as the ones present in figure 3.1 used to cost around 10 times more than a medium-range mobile phone, however, nowadays it is easier to find innovative solutions such as Thuraya SatSleeve [67] that considerably reduce this difference. Nevertheless, this adds difficulties from the logistic standpoint because it is necessary that all users use this kind of device when deployed in a zone with solely satellite coverage. Furthermore, line of sight varies greatly and in some environments, coverage for satellite phone is not powerful enough. Consequently, this architecture is not considered as a valid option.

3.1.2 Temporary Network with Satellite Backhaul

The second option is to relay only in LTE-capable terminals and deploy a temporal LTE access network, a base station, to connect the users located in areas without infrastructure with the rest of the network. This option is simpler for the users and opens the technology to everyone with a mobile phone, which is useful in the cases where multiple organizations collaborate. Figure 3.2 shows this possibility.

In cellular networks, the backhaul refers to the part that connects the access network with the core network entities that manage the mobile network. As we can see in figure 3.2, the corresponding LTE core network entities are located close to the satellite gateway in a zone with dedicated infrastructure. This option would generally need to transfer to IP the flows in the satellite backhaul to use regular satellite terminals and save bandwidth.

In this case, a geostationary satellite will usually provide the connectivity to the temporary cells. Usually, a small satellite terminal will access to satellite capacity on-demand using a return-link standard, like DVB-RCS.

RESTRICTED DOCUMENT

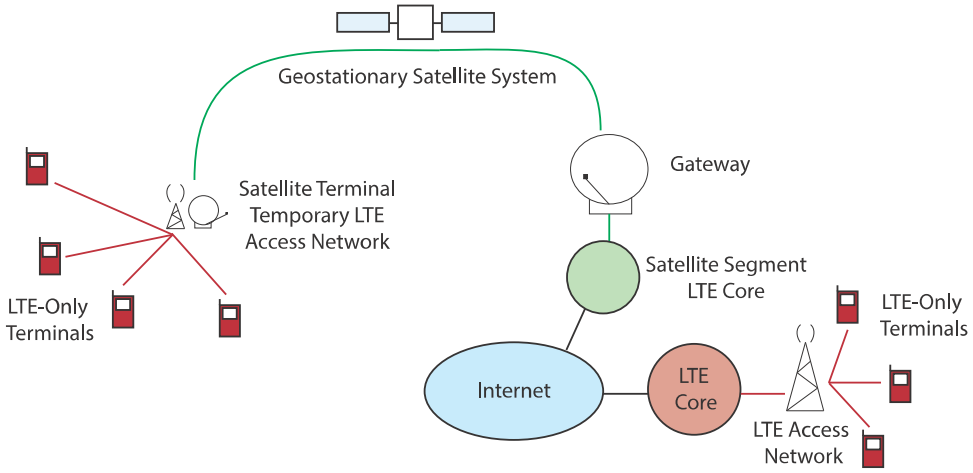


Figure 3.2: Temporary Network with Satellite Backhaul

This architecture is optimal to extent coverage in low-density areas and in emergency situations when the commercial infrastructure might be unavailable or collapsed. The simplicity both user terminals and temporary infrastructure allows a quick deployment.

3.1.3 Temporary Network with Satellite Interconnection

Similar to the previous case, a full LTE network can be deployed in a given area. Even if this network remains temporary, it features a full LTE core, so the output towards the satellite terminal is not a backhaul link anymore but an IP link. Hence, the satellite serves as interconnection between the LTE network and the rest of the Internet. The optimization of the satellite traffic follows the same principals of the fixed satellite systems. Figure 3.3 presents this architecture.

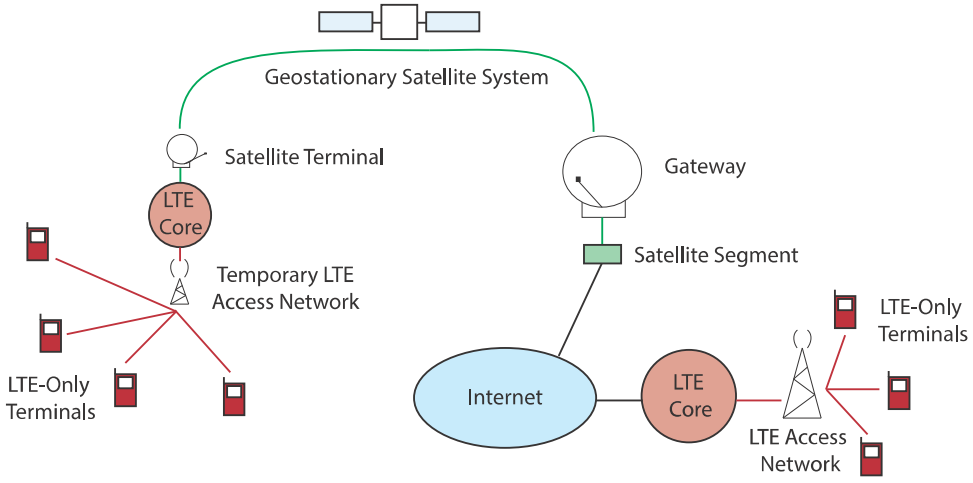


Figure 3.3: Temporary Network with Satellite Interconnection

Thanks to the advance in the equipment manufacturing, it is possible to buy a single equipment that

RESTRICTED DOCUMENT

groups all the entities that manage the core protocols. In addition, larger temporary equipment can also be useful and feature bigger antennas that cover a larger area.

The advantage of this scenario is that it does not need a LTE core available at the other side of the satellite link. The temporary network could operate perfectly on its own to provide communications with the deployed teams. However, the standard covering the eNB functions is evolving and more self-organizing options are being developed.

It is expected that these two last reviewed options will converge over the years. Base stations will be more capable to operate and make their own decisions without the interaction with a LTE core; and the simplification and miniaturization of the core entities will help develop cheaper and smaller cores.

More than one temporary network can be deployed in an area. One possibility is to deploy them close enough to permit the direct communication between them and even relay only on one satellite station for the interconnection or backhauling. Yet the satellite link dimensioning becomes challenging and it is preferred that each temporary network uses its own satellite terminal. The configuration of the different stations is more complex, as they are linked by a multi-hop geostationary link, but as long as the radio interface configuration avoids interferences this option provides a more dynamic deployment.

The options with temporary LTE networks are perfectly oriented to our scenario of reference and are, therefore, the two considered architectures in this thesis.

3.2 PTT Intelligence Positioning

The second part of this architecture choice focuses on the positioning of the Push-To-Talk service decision points. The following options can be incorporated to any of the two hybrid network architectures discussed above.

3.2.1 Full-Distributed

The first option is to let the user terminals coordinate and optimize the push-to-talk communication session. This is the case that is shown in figure 3.4.

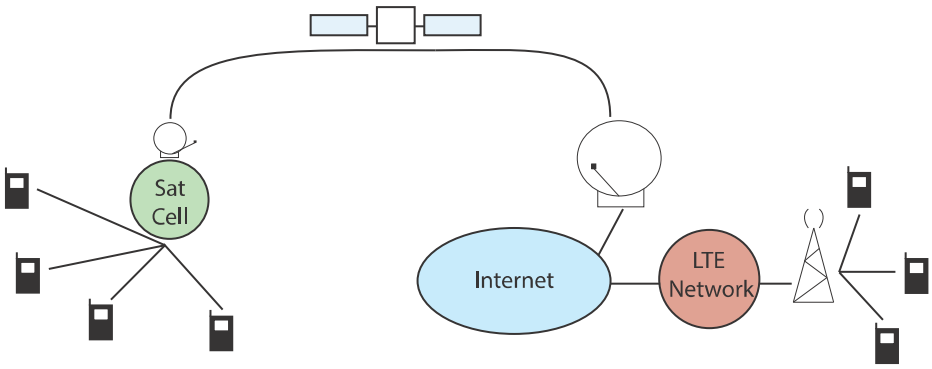


Figure 3.4: Full Distributed Architecture

RESTRICTED DOCUMENT

The service is managed between equal entities and it copes well with the inclusion of the direct mode, the communication between mobile phones without the participation of a network. However, this scenario presents difficulties in terms of scalability and distribution of the service information. At some point, some users could be isolated from the rest without being able to communicate with the rest. Then, regrouping could be complicated.

3.2.2 Centralized

The second case is to introduce a central entity that manages the optimization of the service and the session functions. The user terminal application becomes simpler and all the intelligence is moved towards a hierarchy point as it can be observed in figure 3.5.

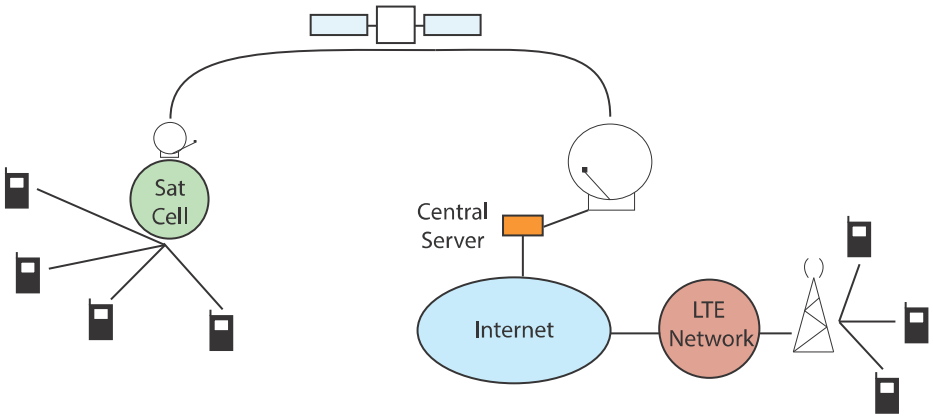


Figure 3.5: Centralized Architecture

This solution, while easy to manage, presents the problem of single point of failure. If the central server cannot be reached, the service cannot operate. This is especially critical in our reference scenario, where the temporary network must be able to manage the groups in isolation. Also, there is the issue of deciding where to locate this central server as it can introduce unfairness between the users of either side of the satellite link.

3.2.3 Semi-Distributed

The last option, the preferred one, is a middle-stage solution. Service management is distributed among several entities, called distributed servers, which can be located closer to the end users while these remain simple. This allows for a better scalability and has not the performance bottleneck issue of the centralized architecture. Figure 3.6 provides an overview of this choice.

This solution permits a balanced hierarchy and enables the possibility to perform reconfigurations into subgroups and to closely manage the deployed users. It introduces also a certain asymmetry in the network allowing each side to proceed independently and synchronize when necessary.

RESTRICTED DOCUMENT

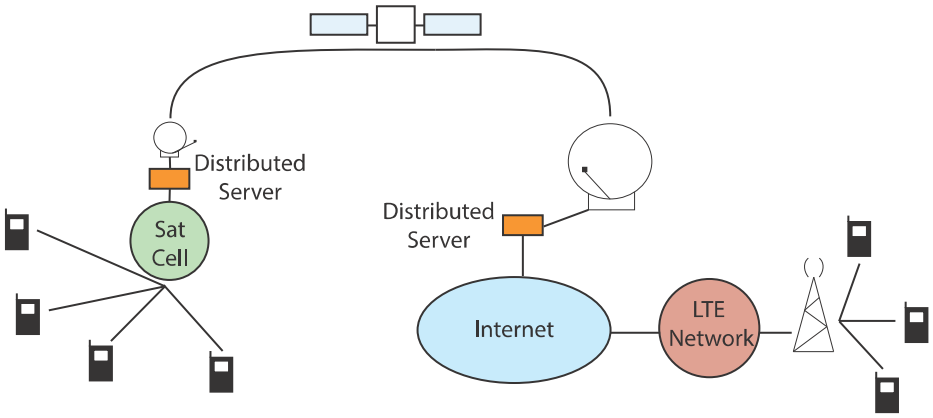


Figure 3.6: Semi-Distributed Architecture

3.3 Final architecture choice

As it can be derived by the previous discussion about the possible architectures, it was decided that the preferred architecture would be a temporary hybrid LTE network with satellite backhaul or interconnection and an organization based on semi-distributed servers. Figures 3.7 and 3.8 show these final options. We have renamed the servers to Super Nodes.

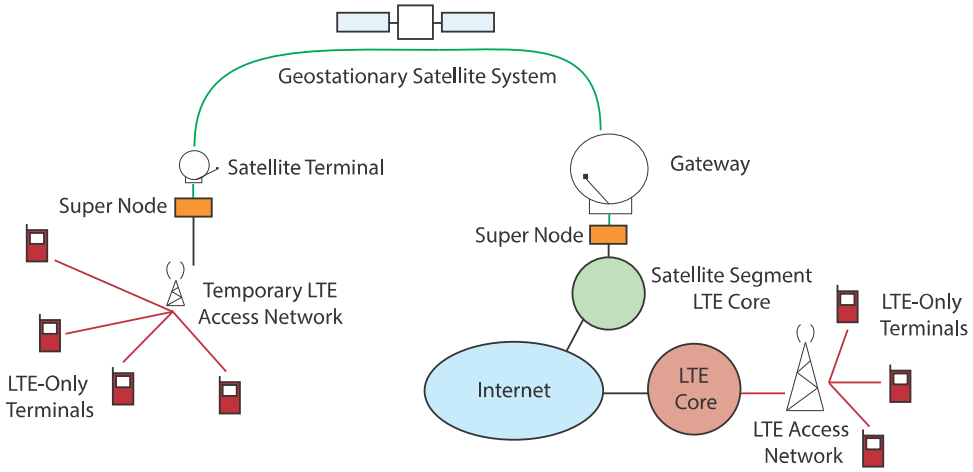


Figure 3.7: Temporary Network with Satellite Backhaul and Super Nodes

The Super Nodes will not only be in charge of the PTT session management and floor control, but they will perform the necessary functions to optimize the network. This is key for the satellite segment where the super nodes can execute roles similar to the ones performed by the performance enhancing proxies. For example, in the case of satellite backhaul, the Super Node will perform the translation of the LTE flows to IP. Further, as we will see in this thesis, they can be involved in the user plane optimization regarding the transmission of voice packets.

The user terminal application becomes simpler in this architecture as all the complex processing

RESTRICTED DOCUMENT

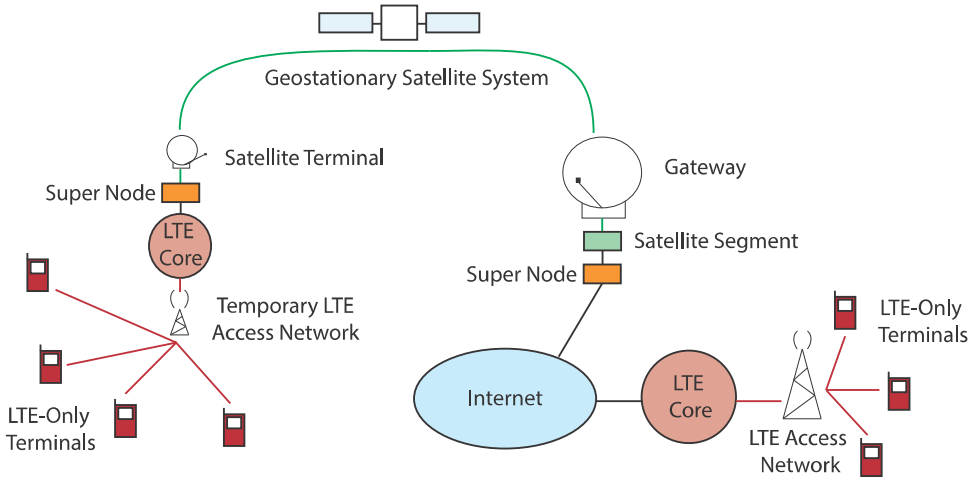


Figure 3.8: Temporary Network with Satellite Interconnection and Super Nodes

decisions are taken at the super node. Additionally, a new user can easily access to the service by simply downloading the user application from its closer super node.

RESTRICTED DOCUMENT

Chapter 4

A Distributed Floor Control Protocol for next generation PMR based on Hybrid LTE and Satellite Networks

4.1 Introduction

In Professional Mobile Radio, floor control regulates the access to shared channels to ensure only one participant is allowed to speak. In the presence of a high delay satellite links and remotely distributed participants, current mechanisms are challenged and fail to take into account the time ordering of the conversations without compromising the overall delay.

In this chapter, we present a distributed floor control protocol tailored to these situations. During the floor determination process, we define ”failure” as giving the floor to a user, while he is not eligible as first user to speak. We derive an analytical model that predicts the probability of failure depending on floor control parameters and the system characteristics. Our simulations provide insight about how floor control should be parametrized so to limit failure probability under a certain threshold.

The rest of this chapter is organized as follows. Section 2 presents the general framework of the floor control protocols. Section 3 provides a survey of previous work on the field of floor control in similar scenarios. Section 4 describes the proposed floor control protocol. Section 5 deepens into the analytical model of the solution. Section 6 reviews the Super Node state machine. Section 7 reviews the requirements of the Public Safety services over LTE and discusses how the presented protocols fits in. Finally, section 8 discusses the experimental methods and results.

4.2 Floor Control

The Internet was initially conceived as a mean for transferring files. In the last decades, it has evolved from a client-server perspective to scenarios where a number of users interact with each other and the

server is only providing the means to make that possible. In this situation, collaborative multimedia applications have emerged. In such applications a number of users from diverse locations share some resources over a network and interact among them. Some examples of this are online gaming, collaborative design or simulation and video or audio conferencing.

Floor control is the set of techniques that help the users to interact seamlessly avoiding any inconsistencies or conflicts. These techniques enable the turn-taking management in a collaborative network. The resource that is being shared is referred as the *floor* and, therefore, floor control deals on who has the right to have the floor at any time of the collaboration session. It coordinates the concurrent access to the shared resources. [68] provides an extensive overview of the different techniques, requirements and classification of floor control protocols.

In push-to-talk solutions, it is the mechanism that manages which client owns the permission to speak, i.e. owns the floor. When multiple users in diverse locations have a conversation, it is likely that the words get mixed and people find difficulties to follow the sense of the discussion. In emergency situations, it is necessary that the exchange of messages goes smoothly and floor control provides help in this direction.

Three types of entities take part in a collaborative session: the Floor holder (FH), the floor coordinator (FC) and the participants. The floor holder is the user that currently has the right to access to the shared resources, i.e. has the right to speak (in a conference-type of application) or to modify the assets (in the case of collaborative design). The floor coordinator is the user or entity that follows a protocol in order to grant the access to the floor. In some cases, the floor coordinator and the floor holder function can be assigned to the same user or node. Finally, the participants are the rest of entities that are listening, viewing or observing in the session. When nobody owns the floor, we say that the floor is idle or free.

A collaborative session pursues a three-step process:

- *Contention phase:* The user requests and gains access to the floor/resource from the arbitrator.
- *Data delivery phase:* The user is able to send data, a message or modify the shared file.
- *Floor release phase:* Finally, when the user finishes, it releases the floor.

Once the user has been given the access to the floor, there are different options leading to a floor release. It can be imposed by the coordinator, signaled by the user himself, enforced after a timer limiting the holding duration expires, or triggered by some other event, like an emergency call request or the entrance of a new user into the session.

Types of floor control

Floor control protocols can be categorized depending on many aspects. For example, depending on the work or presence of the floor coordinator:

- *Assistive:* A moderator facilitates the management of the floor.

- *Autonomous:* Only regular users participate in the floor assignment process, i.e. no user or entity acts as the arbitrator.
- *Automatic:* A non-human agent controls the floors following a set of policies.

It can also be classified depending on the role of the user in the floor assignment procedure:

- *Explicit:* the turn-taking process is pre-defined and enforced.
- *Implicit:* Users have the choice to ask for the floor according to the dynamics of the group.

Further, there can be multiple coordinators or a single one:

- *Centralized:* A single controlling node, a floor control server or a moderator, takes care of the turn assignment. Although this leads to a simple implementation, the central node can become overloaded and its failure can jeopardize the entire session.
- *Distributed:* The control of the floor is shared among autonomous hosts. This approach is more stable, flexible and easier to scale. However, monitoring of the floor control system may become complex. These protocols are similar to the medium access control (MAC) protocols where several nodes/users try to get access to a medium (the floor).

Additionally, floor control can be implemented at the user level without any control message exchange:

- *Received-based:* It is a passive control. The node receives from all sources and filters specifics media streams and ignores the rest. Only the desired streams (from the floor holders) are decoded and played. The user has the control over what it sees/listens. There is a significant network load as all users willing to share keep sending media, so there is a strong need of bandwidth.
- *Sender-based:* Only the floor holder is able to send media to the network reducing the total consumed bandwidth. Switching time between users is higher as it involves entering into a new floor assignment process.

Finally, depending on the number of requests that the protocol is handling at the same time:

- *Non-persistent:* Only first request is kept and the rest are automatically discarded and denied. It is valid for a first-come-first-served policy.
- *Persistent:* A finite number of requests are maintained in a queue. Several policies can be considered to finally decide who takes the floor.

RESTRICTED DOCUMENT

Properties of a Floor Control Protocol

A floor control protocol shall verify the next properties:

- *Correctness:* Each request issued by a user is ultimately served (granted or denied).
- *Promptness:* A request is served with the minimum possible time.
- *Fairness:* All users requesting the floor are equally serviced in average and there is no starvation of sites.
- *Stability:* Control information remains consistent even in the presence of failures, requests are not lost and system must be able to overcome the loss of a node.

Mechanisms versus Policies

Mechanisms regulate the control messages flow and the synchronization across the different users whereas the policies determinate how the floors can be requested and then granted or denied.

Mechanisms are related to mutual exclusion, concurrent control and multiple access technologies:

- *Negotiation:* Floor reassignment is anarchic and decided among all users through an election process.
- *Token passing:* Floor follows a pre-defined hop sequence between the users or the current holder pass the turn directly without any previous request.
- *Token asking:* Users request the floor to the current floor coordinator instead of having an automatic token passing.
- *Time stamping:* Marks each request or message with a global synchronized clock that helps ordering of the events and follow the protocol accordingly.
- *Two-phase locking:* Locks the access to shared documents during the design phase and releases them once they are completed, prohibiting the requests for new locks.
- *Blocking:* Distributed semaphores block the access to other users applying a first-come-first-served ordering.
- *Activity Sensing:* In a contention phase, the users perceive the shared channel activity and back off in case they sense a possible collision between different requests. After a time, they re-try to access the floor.
- *Reservation:* The users have access to the resources during a defined period.
- *Dependency detection:* Attempts to reorder the floor flow events in order to maintain the global causality.

RESTRICTED DOCUMENT

Policies help the floor assignment depending on the implemented mechanism and session parameters:

- *Chair/Moderator:* A specific user acts as the arbitrator over the access to the floor.
- *Agenda:* Floor granting follows an explicit schedule.
- *Time-outs:* Floor requests can be valid for a defined time or the user is allowed to hold the floor during some period and then it must release it.
- *Predefined order:* The floor is granted following a sequence. First-come-first-served would also be considered as part of this policy.
- *Reordering:* The coordinator gathers several requests and serves them depending on the priority or QoS parameters.
- *Election:* Users can vote on who should hold the floor in the next round.
- *Lottery:* Floor assignment is given randomly in a probabilistic scheme.

4.2.1 Objectives of the proposed floor control protocol

In the presence of large propagation delay satellite links (more than 250 ms), current floor control mechanisms are challenged because the location of the participants plays an important role. The clients closer to the moderator benefit from a lower delay and the system becomes unfair. Existing solutions do not provide a method that at the same time maintains a low access time (the time between the moment the user issues the request and the moment he can start talking) and assures that the conversation temporal sequencing is preserved, i.e. the listeners from both sides of a satellite link hear the same conversation, in the same order, helping the coordination of the group. The objective of this contribution is to offer a fair solution that respects the temporal causality for all the participants, independently of their location.

We focus on automatic algorithms, i.e. coordinated by a non-human entity, and on time stamping mechanisms. At first, we do not consider the possibility of having a different access priorities among users and the objective to give access to the first user asking, in a first come first served fashion. Additionally, as we expect to manage users in separate locations, we think a distributed arbitration could be beneficial as we approach the decision makers to the users.

In addition to the diverse classifications we have seen in this section, we add another distinguishing between pessimistic (or non-optimistic) and optimistic protocols. In the pessimistics ones, the coordinating nodes follow a strategy of strict conflict avoidance and only grant access once they are sure that the user is the permitted one. On the other hand, optimistic algorithms allow the presence of some conflicts and provide the means to resolve them. We adopt this last option in order to mitigate the long transit times through the satellite link.

RESTRICTED DOCUMENT

4.3 Previous Work on similar scenarios

Different approaches have been presented in the field of floor control over cellular networks or for interconnecting different types of networks. In this section, we review some of them.

In [69], engineers at Ericsson present a PTT solution with a novel architecture. The floor control server is located at the core of the IP Multimedia Subsystem (IMS). They adopt a centralized architecture where the server could use a first come first serve (FCFS) algorithm. The implementation of the server within the core of IMS allows application developers to exploits its capabilities through APIs.

Another protocol described in [70] highlights the need for a smooth transition between two consecutive users granted to speak. In the process of floor handoff, the second user holds the floor before the listeners have played all the messages from the previous speaker. They receive packets from the new holder while playing the last ones from the previous one, which allows to change of speaker instantaneously.

Authors in [71] propose a full-distributed protocol for a push-to-talk over cellular (PoC) application. Instead of relying on floor control servers, all the users of a group call become floor coordinators. In this approach, a client needs to wait for the confirmation from all the clients before being able to speak.

Floor control protocols are similar to medium access control (MAC) protocols in the way that multiple users compete to get access to a given resource. [72] propose the adaption of some well-known MAC protocols, the distributed queue dual bus (DQDB), ALOHA, and carrier sense multiple access (CSMA) as solutions for the floor control problem over overlay networks. Another work described in [73] also borrows from the activity sensing mechanism of CSMA.

Rohill holds a patent of a protocol that allows to set a PTT conversation between users in different systems [74]. Each user within the same system connects with a given floor control server, similar to our architecture. The different servers use a mechanism that recalls the distributed Performance Enhanced Proxies (PEP), where the server closer to the user requesting the floor supplants the other servers and sends a grant message. In case of several users issuing requests, the servers will compare the sending time. The system does not provide, however, any technique to prevent that the rest of the users, the listeners, see the collisions and receive packets from multiple sources. Cisco also has a system called IPICS that performs like an integrated PEP [75].

Several works propose to wait a given period of time before confirming the permitted client. Upon the receipt of the first request a timer is started and during its duration, another request could be granted instead. The timer is usually the same throughout the set of coordinating servers. [76] proposes a distributed architecture with several severs deployed. They state that their timer lasts for twice the maximum message delay. Additionally, [77] buffers the first data stream from every user requesting the floor. They use a random policy to decide who gets the floor instead of observing the time of issue. In [78], the waiting time is a function of the load of the system. Finally, [79] presents a service for intelligent transportation systems implemented directly on end-users, where no servers take part of the floor determination process. Although the protocol considers forwarding the request through a multi-hop process,

RESTRICTED DOCUMENT

as the vehicles have a limited coverage, some users may be disconnected from the rest.

Our proposed solution reduces the access time, the period of waiting between when the user requests the floor and when it can start speaking, which in some of the reviewed works is at least twice the maximum end-to-end delay between clients. Instead of relying on final users to coordinate the protocol, as a number of the above solutions do, we prefer to keep the client application as simple as possible and rely on a distributed set of floor control servers. We consider that implementing a buffer timer is key to design a protocol over high latency networks. The buffer mechanism reduces the access time and prevents the listeners to see any possible collisions of requests. We go further by adapting its duration to limit the system failure probability depending on the distribution of the requests and the delay between users. Synchronizing the end of the timers increases the fairness of the protocol and allows users to hear the messages at the same time.

4.4 Protocol Description

4.4.1 Protocol overview

We propose a distributed architecture where a set of entities named Super Nodes (SN) manage the floor control protocol. Users are regular LTE phones with an application that connects with a given SN, which is seen as a centralized coordinator by them. In Figure 4.1, we can observe how a simple call between two parties is implemented. During the floor determination process, SNs buffer the voice packets of the users requesting the floor during a period of time before determining who was the true-first (the permitted client).

Upon the receipt of a local floor request from user A, the SN 1 sends a pre-grant message and starts a buffer timer 1, waiting for possible additional requests. The request is forwarded to the other Super Node, SN 2, which starts its own buffer period 2. If another request with an earlier sending time was received, the user issuing it would become the new temporary floor holder. At the end of each buffer timer, the winner is determined and the SN 2 sends the voice messages to the listener, the user B.

The duration of the buffer period is adapted at each SN depending on the probability distribution of the requests and the experienced delay with the requesting user. The objective is to have all the timers expire simultaneously allowing all the listeners to play the message almost at the same time. We define ”failure” as giving the floor to a user, while he is not eligible as first user to speak. When multiple users try to access the floor, an error occurs if the request of the permitted client arrives at the SN once the buffer timer has expired. We derive an analytical model that predicts the probability of failure depending on floor control parameters and the system characteristics. Buffer timers are adapted in order to limit failure probability under a certain threshold.

RESTRICTED DOCUMENT

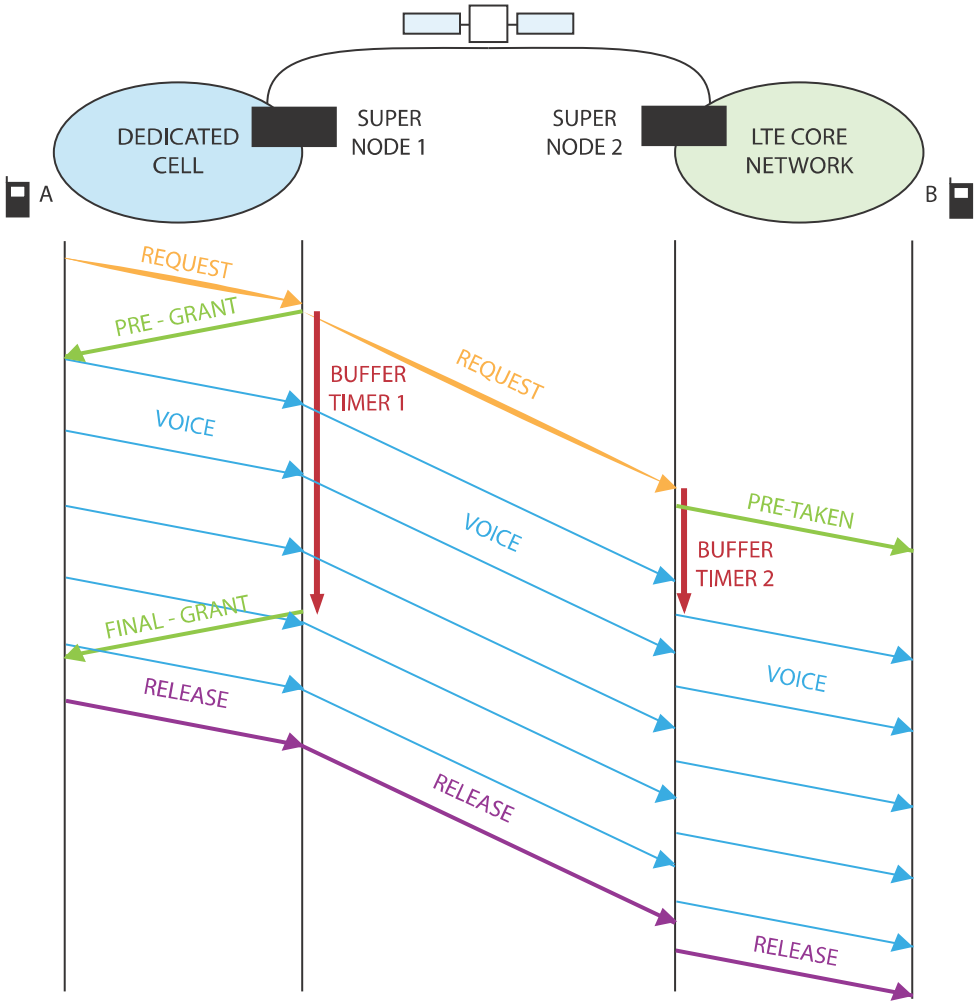


Figure 4.1: Overview of the solution

4.4.2 Protocol Detailed Description

The main objectives of our solution are reducing the access time and assuring that the listeners receive the messages almost at the same time. We define our protocol as optimistic, because it gives access to a requesting user before assuring he will be the final permitted client.

Every request includes a relative timestamp, which defines the period of time between the reception of the last floor release and the moment the new request is sent. A request will be finally accepted only if its timestamp is the smallest among all the requests in the same floor contention iteration.

The users wishing to speak request the floor to their SNs. When a SN receives a local request, it sends a pre-grant message and forwards the request to the rest of SNs. At this moment, the SN calculates the value of the buffer timer and triggers it. From this instant, the user shall speak and its packets will be buffered and forwarded to the remote SNs. These SNs also set a buffer timer when they receive a remote request.

It is possible that during the buffer period another request is received. Then, the SN compares the

RESTRICTED DOCUMENT

two timestamps and if the one from the new request has a smaller value, the user issuing it becomes the new temporary floor holder, the user speaking is notified by its local SN and its messages will not be sent to the rest of users. The buffer duration is then recalculated depending on the parameters from the new request.

Once the timers expire, SNs start emptying the buffer by forwarding the voice packets to their respective listening users. The SN continues buffering the messages from the speaker until the release message is received. The listeners receive the whole message at the same rate it was sent with a time shift given by the buffer period.

The contention period is the time when the users attempt to access the floor. It starts when the first user requests the resources and it ends when its messages are forwarded to the listeners. One goal of the buffer timer is that the different SNs observe a similar contending period. The timer will be shorter in the SNs farther from the requesting users, where the experienced delay is higher. The calculation of the buffer timer will be discussed in the next section.

On one hand, the advantages of such a system is that the requesting user can start speaking quickly once confirmed by its local SN; the listeners do not see the possible requests collisions; the timer also prevents sending messages to the listeners before knowing the final permitted client; and, finally, the buffering mechanism allows forwarding the voice packets quickly. As the timer is adapted depending on the delay, all the listeners will experience a similar mouth-to-ear delay.

On the other hand, the drawbacks are that a user can be cut to give access to another; timers should be optimized to prevent errors and the use of resources increases during contention because all the users requesting the floor will send voice packets before assuring they are the permitted users. When a request with a smaller timestamp arrives once the buffer time is over, the system experiences a failure, because the message from someone else has started to be sent to the listeners. In that case, in addition to cut the user speaking, the listeners see the collision as they hear two different messages.

Figure 4.2 displays an example of system failure at a single Super Node. Super Node 2 receives the request from client A, the final permitted participant, once the timer triggered by the request of client B has ended. Therefore, client B is notified and his message is cut. Finally, users from SN 2 receive the message from A. The objective is to minimize the failure probability and this will be analyzed in the next section. Note that the error may occur in a single SN while the rest could perform well.

4.4.3 Additional options

In this section we discuss about different options that could be incorporated to the protocol.

Instead of answering automatically to a request with a pre-grant message, the SN could wait a short period of time before sending it. Figure 4.3 shows an overview of this possibility where T_{go} accounts for this waiting period before issuing the pre-grant message. This option tries to prevent the loss of the floor because of another request coming from the same SN. The calculation of T_{go} would be similar to the procedure for obtaining T_{buffer}. The difference would come from fixing a higher error probability as target or by considering only the delay of local users, i.e. a different distribution function considering

RESTRICTED DOCUMENT

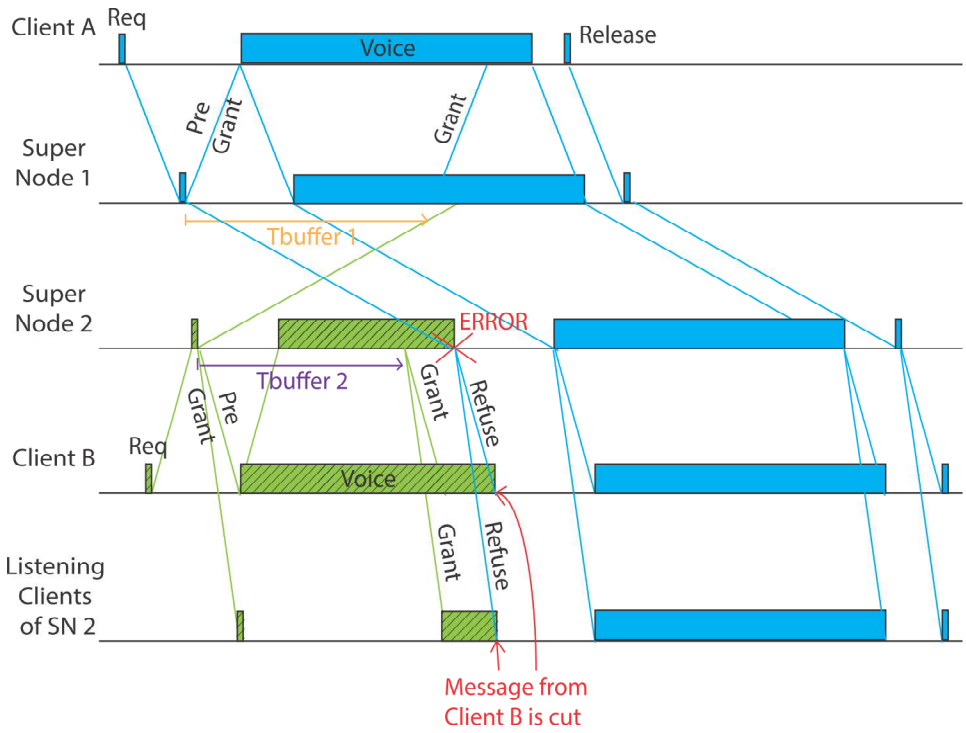


Figure 4.2: Example of system failure

only the local requests. If we considered $T_{go} = T_{buffer}$, the solution would no longer be optimistic and there would not be any buffering period.

The protocol could incorporate an additional call prioritization mechanism with several priority levels. Requests would only compete with others of the same degree. In case a request with higher priority arrives, it overrides the current call and a new contention iteration starts. The SN will buffer the first messages waiting for other possible requests of the same priority rank. If a single user holds the maximum level, a buffer timer is still used to maintain a similar mouth-to-ear delay to all users.

We suppose that the SNs have a mean to synchronize their clocks, such as Global Positioning System. The synchronization between a given SN and its clients is performed at a given frequency, trying to limit the possible issues. If the access networks from users to SNs are similar throughout the network, one could use only the values at the SNs, considering a delay of 0 between a SN and a local user. The T_{go} mechanism would be necessary to limit the probability to cut a user message.

4.5 Analytical Model

4.5.1 System failure probability

The aim of this analytical model is to derive the duration of the buffer timer (T_{buffer}) that maintains the failure probability under a given threshold. A failure occurs when the request from the permitted client arrives once the buffer period is over. Mathematically, the condition to experience a failure can be

RESTRICTED DOCUMENT

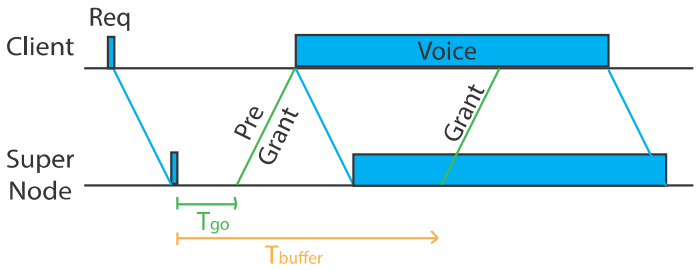


Figure 4.3: Overview of T_{go} for a retarded pre-grant

expressed as follows:

$$T_1 + d_1 + T_{buffer} < T_2 + d_2 \quad \& \quad T_2 < T_1 \tag{4.1}$$

Where T_1 and T_2 are the timestamp from the first and the second requests received; d_1 and d_2 are the experienced delay of these two requests, i.e. the time between the moment the user sends the request and the SN receives it; finally, T_{buffer} indicates the duration of the buffer period.

When the SN receives the first request, it knows the values of T_1, d_1 and their sum. Then, the buffer duration is calculated assuming that $T_1 + d_1 < T_2 + d_2$, i.e. the request was received before any other. Hence, we derive the probability of system failure at a single SN when two users are contending for the floor, $P_{failure\_one}$:

$$P_{failure\_one} = P\left\{T_1 + d_1 + T_{buffer} < T_2 + d_2 \quad \& \quad T_2 < T_1 \quad \| \quad T_1 + d_1 < T_2 + d_2\right\} \tag{4.2}$$

Applying the theory of the conditional probability of Bayes, we have:

$$P_{failure\_one} = \frac{P\left\{T_1 + d_1 + T_{buffer} < T_2 + d_2 \quad \& \quad T_2 < T_1 \quad \& \quad T_1 + d_1 < T_2 + d_2\right\}}{P\left\{T_1 + d_1 < T_2 + d_2\right\}} \tag{4.3}$$

Observe that, since $T_{buffer} \geq 0$, the joint probability can be simplified because the event of $T_1 + d_1 < T_2 + d_2$ is a special case of $T_1 + d_1 + T_{buffer} < T_2 + d_2$. Additionally, the event of $T_1 + d_1 < T_2 + d_2$, may occur when $T_1 < T_2$ or $T_1 > T_2$. Therefore, the final expression of $P_{failure\_one}$ can be written as:

$$P_{failure\_one} = \frac{P\left\{T_1 + d_1 + T_{buffer} < T_2 + d_2 \quad \& \quad T_2 < T_1\right\}}{P\left\{T_1 + d_1 < T_2 + d_2 \quad \& \quad T_1 < T_2\right\} + P\left\{T_1 + d_1 < T_2 + d_2 \quad \& \quad T_1 > T_2\right\}} \tag{4.4}$$

4.5.2 Parameter characterization and estimation

In order to calculate $P_{failure\_one}$, it is necessary to characterize the different parameters of the system. First, the call idle time, the period between the receipt of the floor release and the moment when the floor is requested, which corresponds to the request timestamp. Second, the same applies to the network delay: the difference between the moment the request is sent and the instant when the SN receives it.

RESTRICTED DOCUMENT

We propose two probability distribution laws to characterize the delay and the call idle time according to the state of the art. If another distribution results in a better fit and characterization of these parameters, its function can be used when calculating the failure probability without changing any other element in the protocol.

Several studies have been conducted to examine the behavior of the PTT traffic. At the beginning, a model with exponential distributions and infinite population with Poisson arrivals, similar to the one used for regular voice calls, was adopted. However, by assuming infinite population, it would seem that more radio-channels were needed to meet the QoS requirements. Models such as finite population or unbalanced multi-population fit better the PMR systems. Recent works, such as [80], show that the statistics of idle and holding times vary significantly over time and demonstrate daily periodicity requiring non-stationary models. Observing long-term statistics, the log-normal distribution best fitted all the parameters.

Hence, we adopt the log-normal distribution as the best choice to characterize the call idle time. The probability distribution function of the random variable, τ, that represents the call idle time is expressed by:

$$f_\tau(t) \quad = \quad \frac{1}{t\sqrt{2\pi}\sigma} e^{-\frac{(\log(t)-\mu)^2}{2\sigma^2}} \tag{4.5}$$

Here, μ and σ, mean and standard deviation of the logarithm of the variable, denote the parameters of the distribution. The parameters are estimated using the likelihood maximization algorithm.

The study of the network delay can be very useful to examine the performance of the system and adopt the necessary changes to optimize it. However, the analysis of such key element remains difficult to be characterized by a single random distribution. Authors in [81] propose a simple and flexible method using a finite combination of Weibull distributions. The probability distribution function of a mixture is a weighted sum of the functions of the different elements that compose it:

$$f_\delta(d) \quad = \quad \sum_{i=1}^{I} q_i * f_{\delta_i}(d) \tag{4.6}$$

Where δ represents the delay random variable, δ_i is each component of the delay and q_i are the weight of each component, knowing that $\sum_i^I q_i = 1$. We extent their approach by using three-parameters Weibull distributions instead of the original two-parameters distributions. One of the main components of the delay comes from the transmission and propagation delay, which are mainly constant. Therefore, the addition of a location parameter (l_i) that accounts for the constant part of the delay improved the overall model performance. The rest of the model parameters are related to the average peak or scale (r_i) and the tail behavior or shape (s_i). The distribution function of each component is defined by:

$$f_{\delta_i}(d) \quad = \quad \frac{s_i}{r_i}\left(\frac{d-l_i}{r_i}\right)^{s_i-1} e^{-\left(\frac{d-l_i}{r_i}\right)^{s_i}} \tag{4.7}$$

The set of $4*I$ parameters is obtained with an expectation maximization algorithm. A good option is to reserve a different subset of distribution components to the messages coming from each Super

RESTRICTED DOCUMENT

Node. This is an easier manner to classify the packets and compute the constant delay of the link that interconnects the SNs.

4.5.3 Failure probability with multiple competitors

Generally, apart from the first received request, more requests, one for each contender, could be received. For a group of N users, assuming that everybody tries to get the floor, the total failure probability would be given by:

$$P_{failure} = 1-(1-P_{failure\_one})^{N-1} \tag{4.8}$$

This is, however, the worst case scenario. Generally, not all the users request the floor. If we are able to obtain the probability of requesting the floor, P_{push} and we assume a binomial distribution, the total failure probability is revised:

$$P_{failure} = 1- \sum_{k=0}^{N-1} \binom{N-1}{k} P_{push}^{k}(1-P_{push})^{N-1-k}(1-P_{failure\_one})^{k} \tag{4.9}$$

P_{push} can be calculated by maintaining a counter for the number of requests received for each floor determination iteration, $requests\_received_m$. At least one request is received every iteration. P_{push} is obtained with the average of all iterations, M:

$$P_{push} = \frac{\sum_{m=1}^{M} \frac{requests\_received_m - 1}{N-1}}{M} \tag{4.10}$$

4.5.4 Buffer timer calculation

Given the complexity of the probability distributions that define the system, there is no closed-form solution (it cannot be expressed analytically in terms of a finite number of certain ”well-known” functions). Hence, numerical methods are necessary to retrieve a final result.

The on-line/real-time computation of the buffer is therefore unfeasible because it could take more time than the real timer duration, depending on the Super Node computation capabilities. As the system behavior remains stable during a period of time, buffer periods are calculated off-line and stored for subsequent use. The integrals to calculate the set of probabilities stated previously can be found in the annex of this chapter.

$P_{failure}$ as a T_{buffer} function is strictly decreasing, i.e. as T_{buffer} increases, $P_{failure}$ decreases. The goal is to calculate the minimum T_{buffer} that maintains $P_{failure}$ under a target defined by the system administrator.

The function of T_{buffer} to meet a given $P_{failure}$ is a function of the call idle timestamp and the delay, the retrieved parameters from each request. A matrix of T_{buffer} values is calculated in a frequency

RESTRICTED DOCUMENT

defined by the supervisor. Once a request is received, the true T_{buffer} is calculated by taking the pre-computed values stored in this matrix and applying a bilinear interpolation. These pre-computed values are obtained using the bisection method, calculating the failure probability for different T_{buffer} values until finding the one that yields a $P_{failure}$ close to the defined threshold.

The entire process for retrieving the T_{buffer} values is detailed next:

1: Estimate the call idle time parameters for $f_\tau(t)$. The likelihood maximization algorithm is used in the lognormal case.
2: Estimate the delay parameters for $f_\delta(d)$. The expectation maximization algorithm is used in the case of a mixture of three-parameter Weibull.
3: Set the number of clients N.
4: Calculate P_{push}.
5: Set the minimum and maximum values of the T_s and d.
6: Set the calculation step for T_s and d. It can be different for each parameter.
7: Set the minimum and maximum values of the T_{buffer}.
8: Set the target for the failure probability, $P_{failure\_target}$.
9: Set the maximum error when calculating $P_{failure}$.
10: Create the matrix of T_{buffer} values. For each couple T_s and d with fixed N and P_{push}, find the T_{buffer} value with a $P_{failure\_result}$ that yields $|P_{failure\_result} - P_{failure\_target}| \leq max\_error$ using the bisection method.
11: Upon the receipt of a new request, the actual T_{buffer} value is obtained using the matrix of values and a bilinear interpolation.

4.5.5 Scalability

One of the main concerns when designing a communication protocol is scalability. Considering a single talk group, an increase in the number of users makes the computation of the formulae remain stable, except if we consider the formula that takes P_{push} into account. The worst case scenario can be used for quicker computation. Nevertheless, administrators should limit the number of users N used in the calculations even in groups with very large number of users. One should roughly estimate how many users will attempt to access the floor at the same iteration, a number that will probably be less than 10.

If we observe a high number of local requests that make the SN modify the pre-granted user, we could use the option of the delayed pre-grant message, discussed previously.

Yet, a growth in the number of SNs implies a higher complexity in the overall computation of the buffer periods. We suggested to treat the requests coming from each SN independently to limit the complexity of the estimation algorithm of the delay distribution. Hence, we observe that the increase of the load is linear. Note that the complexity of the integrals to solve is only increased by the number of summands.

In the case of several talk groups, one could treat them totally independently or aggregating all the requests coming from the same SN. In that way, the estimation of the parameters of the delay could be

RESTRICTED DOCUMENT

the same for a number of groups. Still, the weighting parameters, the $q's$ in the presented distribution mixture, should be tuned for every group in order to represent the percentage of the messages coming from each SNs. Hence, delay estimation is performed in two stages: first, the estimation of the parameters of the delay for requests coming from a given SN; and second, the final $q's$ are obtained by multiplying the obtained $q's$ in the first step by the percentage of requests coming from this SN in the given group.

Finally, we expect that the deployed cells or SNs, the ones that need to be back hauled, manage a small number of talk groups and users, whereas a control center could face a much higher load, while benefiting from a better hardware to handle it.

4.6 State Machine

In this section we review the floor control state machine from the Super Node standpoint. Figure 4.4 shows the different states and the possible transitions.

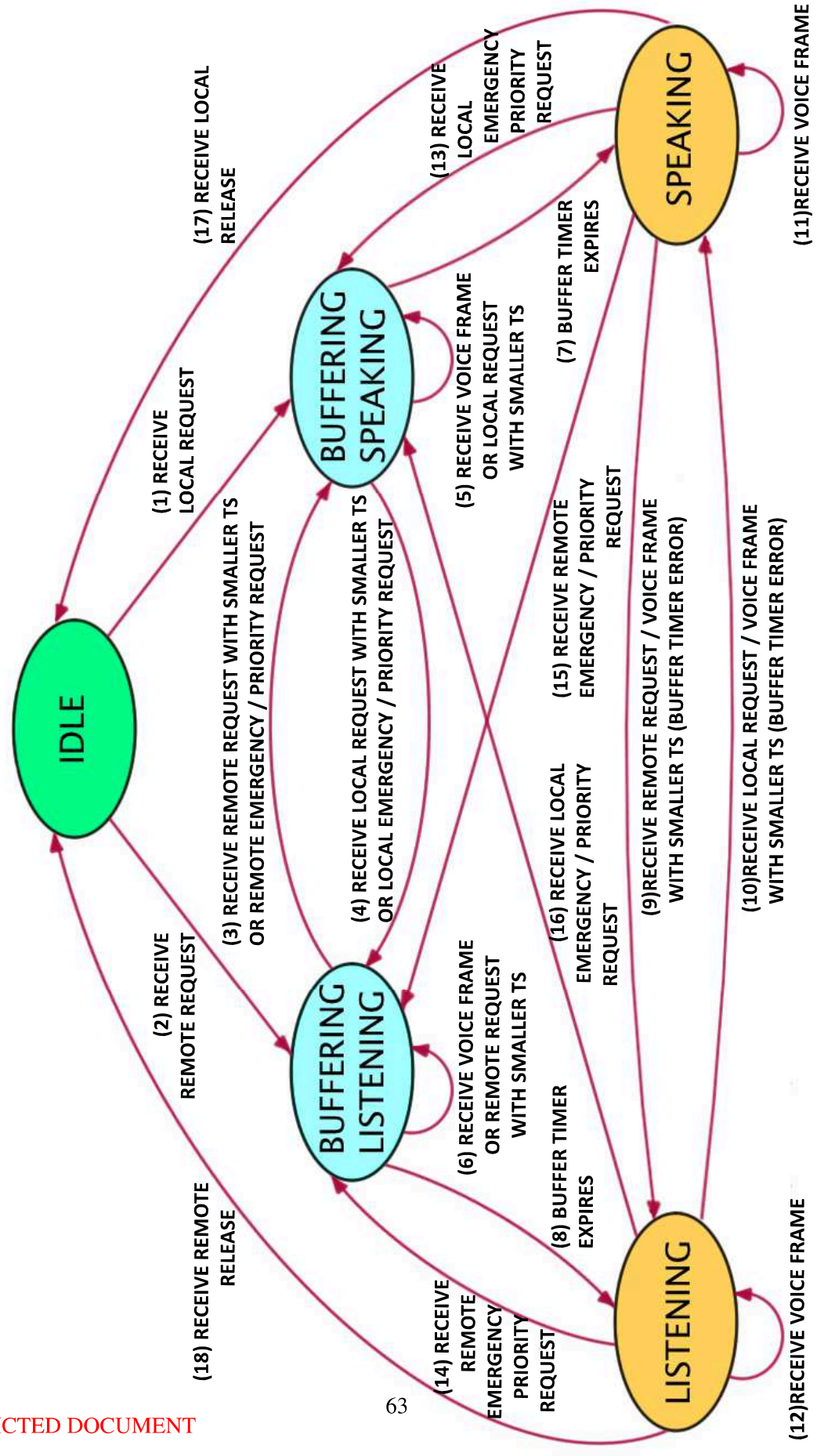
The SN can change between five states:

- *Idle:* There is no activity on the concerned group.
- *Buffering Speaking:* The request with the smaller timestamp yet received is from a local user. The SN is buffering the voice frames coming from this user.
- *Buffering Listening:* Similar to the previous one but the temporarily permitted user depends on another SN.
- *Speaking:* The floor has been granted to a local user after the buffer timer has expired. The user is talking.
- *Listening:* The permitted user is managed by a remote SN.

Next, we review the possible transitions between the states. Transitions have been numbered in figure 4.4 for better understanding.

1. *Idle* $\rightarrow$ *Buffering Speaking:* The SN receives a request from a local user and it sends a pre-grant.
2. *Idle* $\rightarrow$ *Buffering Listening:* The SN receives a request from a remote user.
3. *Buffering Listening* $\rightarrow$ *Buffering Speaking:* The SN receives a request from a local user. As it already received another request, it compares the timestamps and concludes that the local user should be pre-granted.
4. *Buffering Speaking* $\rightarrow$ *Buffering Listening:* The SN receives a request from a remote user. The previously received request is rejected as the newer has a smaller timestamp.

RESTRICTED DOCUMENT



RESTRICTED DOCUMENT

5. *Buffering Speaking → Buffering Speaking:* A voice frame from the temporarily permitted user is received. The SN also remains in the buffering speaking state if another local request with smaller timestamp is received and if a request that should not be granted is received, regardless of its provenance.

6. *Buffering Listening → Buffering Listening:* Similar to the previous one but when the provisionally permitted user is a remote user or if a remote request with smaller request is received.

7. *Buffering Speaking → Speaking:* The buffer timer expires and the voice frames can be forwarded to the rest of participants.

8. *Buffering Listening → Listening:* The buffer timer expires and granted user is remote. Local users start receiving the message.

9. *Speaking → Listening:* A local buffer timer error has been experienced. A request from a remote user with smaller timestamp is received when a local user was talking and its message will be cut.

10. *Listening → Speaking:* Similar to the previous one. A remote user was talking and a local request with smaller timestamp has triggered a buffer timer failure. If the buffer timers are well dimensioned, this transition is unlikely to happen.

11. *Speaking → Speaking:* A local user is talking and its voice frames are forwarded to the rest of participants.

12. *Listening → Listening:* A remote user is talking and the local users are listening to the message.

13. *Speaking → Buffering Speaking:* An emergency request or a request with higher priority from a local user is received. The contention phase is restarted.

14. *Listening → Buffering Listening:* An emergency / higher priority request from a remote user is received while another remote user was speaking.

15. *Speaking → Buffering Listening:* An emergency request or a request with higher priority from a remote user is received. A local user was talking and is cut.

16. *Listening → Buffering Speaking:* An emergency / higher priority request from a local user is received while a remote user was speaking.

17. *Speaking → Idle:* The local user talking indicated that he has finished and sends a floor release. It is possible that some voice frames have not been forwarded to the listening participants yet.

18. *Listening → Idle:* A floor release from the remote user speaking is received. The SN will finish sending the voice frames and inform the local users that the floor is free.

RESTRICTED DOCUMENT

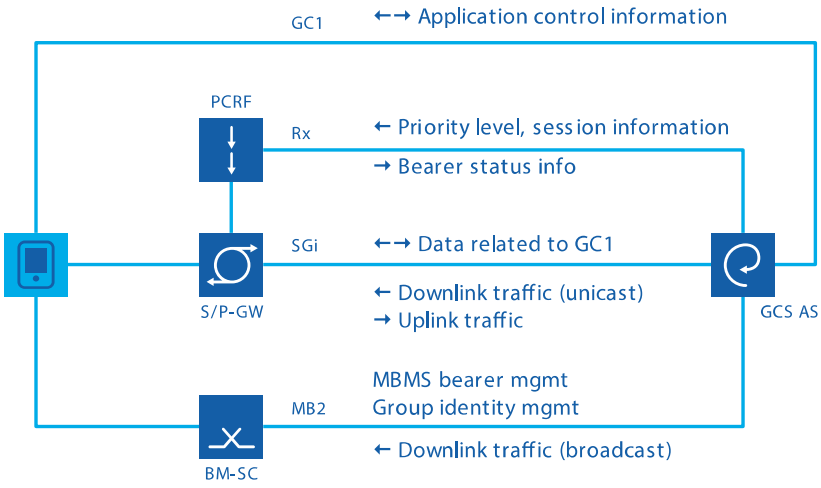


Figure 4.5: Group Communications architecture [10]

4.7 Compliance with the evolution of the LTE standard

Since the decision to adopt LTE as the driver technology for new generation Public Safety networks, 3GPP has been working on a series of specifications as we commented in chapter II. In this section we discuss how the proposed solution can be incorporated in this framework. Basically, our interest is to review the specifications concerning the requirements of the Mission Critical Push-To-Talk service [17] and the management of group communications [51].

A Super Node, in the sense we described it, is an Application Server in the overall architecture of the group communications over LTE, denoted as GCS AS (Group Communication Service Application Server). As it is located in the application domain, it is not associated with a given access network. The LTE architecture sees it as a third party application servers. Application signaling and group management aspects are out of the scope of [51], the specific duties of the SNs.

Communication from the GCS AS towards the users can be performed either through unicast bearers or using multicast/broadcast bearers from the eMBMS service. The GCS AS should select the preferred method and establish the corresponding bearers. Bearers of different kind can coexist for the same group depending on the characteristics and location of the participants. Evidently, unicast is the only possible method for the uplink between users and the server. Figure 4.5 shows the high-level architecture of the group communication service as well as the information flows.

The SN needs to ask for resource allocation when it first receives a request. To set up a new bearer, it first needs to register the group filing an ”Allocate TMGI (Temporary Mobile Group Identity) request” to obtain an identification and then a demand to activate the MBMS bearer request. It is expected that call setup will delay the final transmission towards the participants. This request for allocation can be performed during the buffering period, reducing the end-to-end setup time. While the final user could change, the process is entirely transparent for the BM-SC. At the same time, it is necessary to set up the bearers for the users that are contacted with unicast.

RESTRICTED DOCUMENT

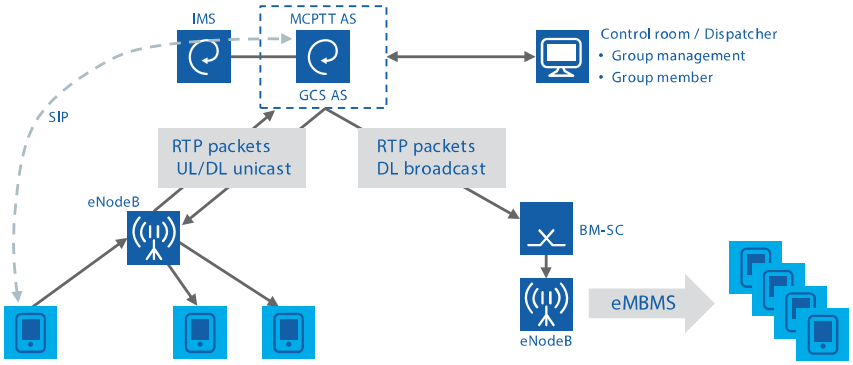


Figure 4.6: Mission Critical Push-To-Talk [10]

At the end of a call, a conversation, the MBMS bearer would be freed up to let other groups take the resources. However, the TMGI allocation would be long term, as the users monitor the MBMS channel for new activity. [82] discusses different alternatives to manage the resource allocation request process in such scenarios.

MC-PTT [17] sets a series of requirements for the PTT service in the LTE architecture. Some of them are specifically related to the Floor Control protocol and other generalities of the PTT service. Figure 4.6 provides an overview of the service architecture which is managed by the an Application Server in the way we saw above and makes use of both unicast and broadcast mechanisms.

MC-PTT main purpose is to provide an arbitrated method to engage communication, mainly through voice, for groups as well as for conversation between two peers. This service will leverage upon the evolution of group communications and Proximity Services in LTE. ProSe [52] aims to provides the means for off-network communication, which is not envisaged in the discussed operation of the protocol.

This arbitrated method is possible thanks to the floor control protocol that has to be deterministic. Our proposal is deterministic in the sense that, given the same priority between users, the user that emits the request with the shorter call idle time will eventually get the floor. Next, we discuss some of the points that are reviewed in the MC-PTT specification.

Call commencement and termination A call is a succession of talk bursts that form a conversation between two or more users. It order to start a call some conditions need to be meet, like the availability of resources and a minimum of participants depending on the nature of the call. Resources in the LTE domain are provided by bearers. As we discussed, the SN, the application server, needs to make sure those are available. In the case of a deployed satellite cell, it will also verify that the satellite return channel can provide the means for the call. This conditions are not checked again for the individual transmissions, therefore, it is expected that the call setup time will be larger that the one for those individual talk bursts. The floor control solution was specially conceived to deal with the arbitration of the permission to transmit for those transmissions. A contention between multiple users at the start of a call is expected to be unlikely.

RESTRICTED DOCUMENT

Resources can be reserved during the duration of the call with the successive transmission. A call will end after a configurable period of time called hang time. We suggest that the protocol should embrace this limitation and not compute buffer timer duration for values of the call holding time greater that the hang time. At the start of a new call, the value of the hang time will be used to determine the buffer timer.

It shall be possible to limit the duration of talk bursts. We denote this as the maximum call holding time and should be configurable. In PMR systems, it is called ToT (Time-Out Timer). In the case it is not set to infinity (no limit), the user application shall indicate the user that the transmission is approaching its end and then cut the microphone and send a release. This limitation has no impact on the analytical model of the solution.

Additionally, an authorized user shall be able to terminate the current transmission. A force release message is sent to the SN, which should communicate the user speaking that his permission has been revoked. The forwarding of the call is not stopped until the last packet from the user arrives or another call enters the system and resources are needed.

Handling and queuing of requests The specification requires that each request should get an answer: it should be granted, rejected or queued. We introduced the concept of pre-grant in order to reduce the access time, we believe it is a necessary tradeoff to improve the end-to-end latency. A refuse or a reject message will be sent to a user that cannot get the floor for the current iteration and when queuing is not enabled or there is not enough space in the queue.

Initially, the queuing of requests was not foreseen in our solution, however, it could be added with ease. The requests will be queued in the order of priority in the first stage and then by length of the call idle time. When multiple calls are concatenated, we should mark the moment of initiation of the call idle time. This would be end of the last received transmission, the release reception. Hence, the order will be provided by the floor control iteration. The final order or the queue should be set by: priority, iteration (which actually relates to the send time) and call idle period. The first request in the queue will be pre-granted after the transmission of the release message to the participants.

In case there were no resources to handle a call, it should be possible to queue the call. Here the priority is first indicated by the group priority and then by the user priority. FIFO queuing should be performed for cases with the same level of priority. In this case, when a call reaches its hang time, the dequeuing of a request occurs.

It is possible that the queues are not consistent between SNs, therefore, they shall be retransmitted towards the receiving SNs to indicate the start of a new call. Notice that buffering will apply as well for the queued requests.

Priority Model The priority in a MC-PTT service is situational and it is intended to give a real-time priority experience. For this reason, we decided to adopt the call idle time as a priority parameter. However, there exists other priorities in the operation environment of Public Safety organizations. Priority is given to users in higher positions in the hierarchy scale and to groups depending on the situation. Gener-

RESTRICTED DOCUMENT

ally, groups englobing multiple teams will have a higher priority and, for example, if the first responders are dealing with a forest fire, the firemen groups will have a higher priority.

First of all, priority is necessary to acquire the resources at the bearer levels which will be first denoted by the emergency situation, then by the group category and finally by the priority of the user initiating the call. Second, at the floor control level, there is a management within the group given mainly user hierarchy. Private calls, with or without floor control, have the priority of the user initiating the call.

Consequently, the first stage is provided by the EPS admission and scheduling controls that will help establish the bearers for the new entrants and pre-empt a current call if necessary. Then, the floor control applies in the context of a single call. We discussed that, in our opinion, the arrival of a request from a user with higher priority during the transmission of a call will indicate the start of a new contending period. Only requests with the same level of priority (in order of emergency situation, group, user and call idle time) can compete.

The specification indicates that there are several possibilities for the ongoing calls fate. The service shall allow the possibility of maintaining a call even if a new one with higher priority enters the system. This means that both overriding and overridden calls can coexist as long as users are well informed of the situation. It is then the turn of the participants who can decide which call to listen to. In the case the configuration prevents this situations or there are not enough resources for the two calls to coexist, the call with inferior priority shall terminate.

MC-PTT specifies two call types with specific higher priority: emergency and imminent peril calls, which can be for a group or private. Upon the initiation of one of these situations, the resources and the priority level shall be maintained until the user cancels this exceptional state.

An emergency call indicates the immediate need of assistance because of a life-threatening situation of the user emitting the request. An imminent peril call differs from the previous case based on whom the help is needed. In this second case, the support is required for other persons or users. Therefore, the final order of priority for situation is: emergency, imminent peril, broadcast and regular call. Note that we included broadcast calls where a specific group is dynamically created within a defined geographic area.

Receiving from multiple calls Users can receive from multiple groups they are affiliated and then choose which one to listen to. This is specially useful for monitoring purposes. Floor control deals with the transmission arbitration, so as long as the resources exist to convey multiple calls to a user, it has no impact to the floor control because the user is not a contender.

Private calls They allow a one-to-one communication without the intervention of a group. They can use floor control or not. In the latter, they will operate as regular full-duplex VoIP calls.

In case floor control is used, the protocol explained will operate as usual, using the values obtained from other calls and groups for the parameters characterization. When only a SN is dealing with the private call, the buffering is expected to be short and will not have much impact in the setup time. When

RESTRICTED DOCUMENT

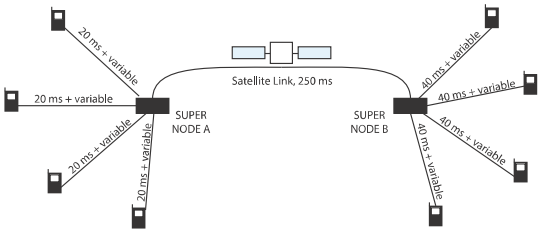


Figure 4.7: Simulation scenarios 1 and 2

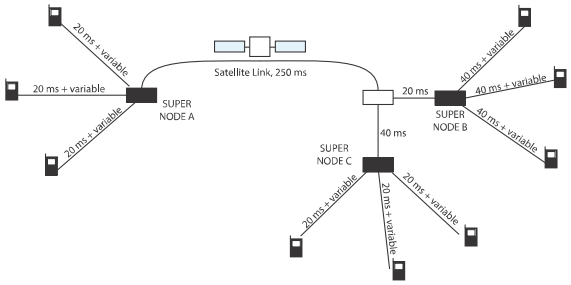


Figure 4.8: Simulation scenario 3

two SNs are dealing and there is a sufficient large delay between them, the buffering will most likely have no impact because the transmission will have already started towards the receiving SN.

Late call entry A member of a group shall be able to join an ongoing call. The user is registered into the SN and this will indicate him how to receive the current transmission or establish a new unicast bearer. The requirements in the specification only apply when users are under the coverage of the same network. This is the case where a SN is already managing the call. We expect that the requirements in this clause will be met.

4.8 Evaluation

4.8.1 Validation of the $P_{failure}$ formula

First, MATLAB™ was used to validate the computation of the failure probability and to observe the evolution of T_{buffer} to meet a given $P_{failure}$. Montecarlo simulations were executed to confirm the process of calculating the set of integrals and obtaining the values of T_{buffer} via the bisection technique.

Given a couple of delay and call idle values, we first computed the T_{buffer} that makes $P_{failure}$ approach to a given threshold. Then, we generated a large number of couples of delay and call idle values that account for possible requests received after the one that is described by the first couple of values. Then, we computed the number of failures the system would have experimented if the SN would have waited the precomputed T_{buffer} before receiving the rest of requests. Finally, we divided this result by the total number of generated requests. The resulting $P_{failure}$ met the desired target validating the presented formulae.

We focused our evaluation on three different scenarios. In the first two, two Super Nodes are connected by a satellite link as we see in figure 4.7. This exemplifies the case with a satellite cell deployed on the emergency scenario connecting to the core network. The difference between these two scenarios is the message distribution. In the first one, users of both sides emit the same quantity of requests, so the conversation has a $50-50$ share between the Super Nodes. The second case is an unbalanced group, where 75% of the calls come from users in the satellite cell. Finally, in figure 4.8 we observe the third scenario with three SNs. This could represent a case where an extra Super Node manages the users

RESTRICTED DOCUMENT

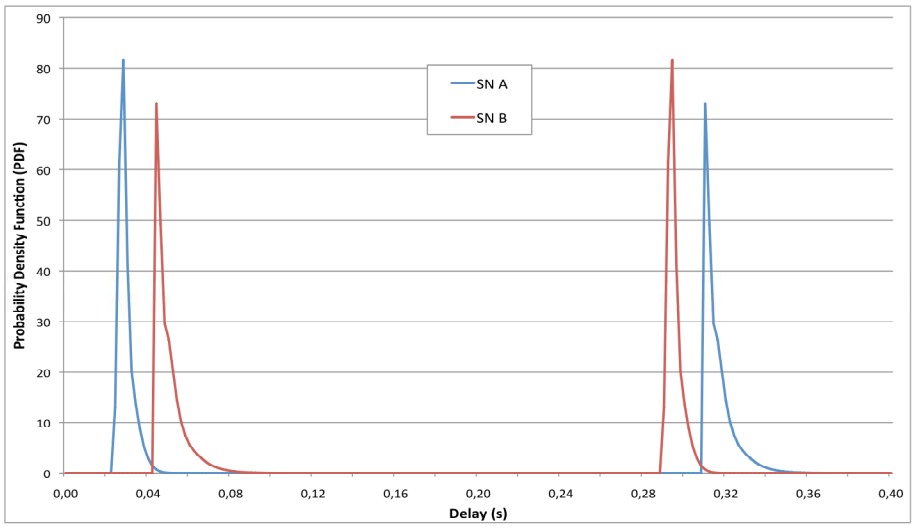


Figure 4.9: PDF of the delay. Scenario 1

located in a safe zone where, for example, the campaign hospital has been installed. The call share for this last scenario is 40% for the users belonging to SN A, 20% for users in SN B and 40% for the rest of users managed by SN C.

It is interesting to observe the resulting density probability function of the delay estimated for each of the SN in every scenario showed in figures 4.9, 4.10 and 4.11. We have left a constant delay between Super Nodes, but in each one, the delay with its users responds to a mixture of three Weibull variables, whose values are provided in table 4.1. The delay parameter for each Weibull variable that the SN will estimate corresponds to the delays observed in the figures showing the scenarios from each SN, i.e. they vary depending on which SN estimates them. The transmission delay was not shown in the scenarios figures. The final values of the q parameters are obtained by multiplying the ones in the tables with the corresponding share of calls depending on each scenario. We include also the values of the call idle time distribution we used and figure 4.12 shows its probability density functions.

Figures 4.13, 4.14 and 4.15 show a series of graphics representing the variation of the $P_{failure}$ depending on the T_{buffer}, the call idle time and the system delay for the first scenario calculated by the SN A. As expected, the $P_{failure}$ decreases as T_{buffer} increases. Note that we do not get a small $P_{failure}$ until T_{buffer} has a value almost equivalent to the actual inherent and constant delay of the network. Considering the evolution versus the call idle time, it is an increasing function at the beginning because it strongly depends on the distribution of the call idle time, as the probability of having a call idle time smaller than 0.5 seconds is very low. However, as T_s increases, $P_{failure}$ remains almost constant because it is the distribution of the delay that drives the function. To conclude, the case considering the delay follows a decreasing function. In fact, the sum of the experienced delay and T_{buffer} is what influences the most.

We show two graphs for the other scenarios, representing the $P_{failure}$ depending on the T_{buffer} in figures 4.16 and 4.17. Comparing the first and second scenarios, we observe that the $P_{failure}$ is smaller in the second case with the same pair of delay and call idle time values because the probability of having

RESTRICTED DOCUMENT

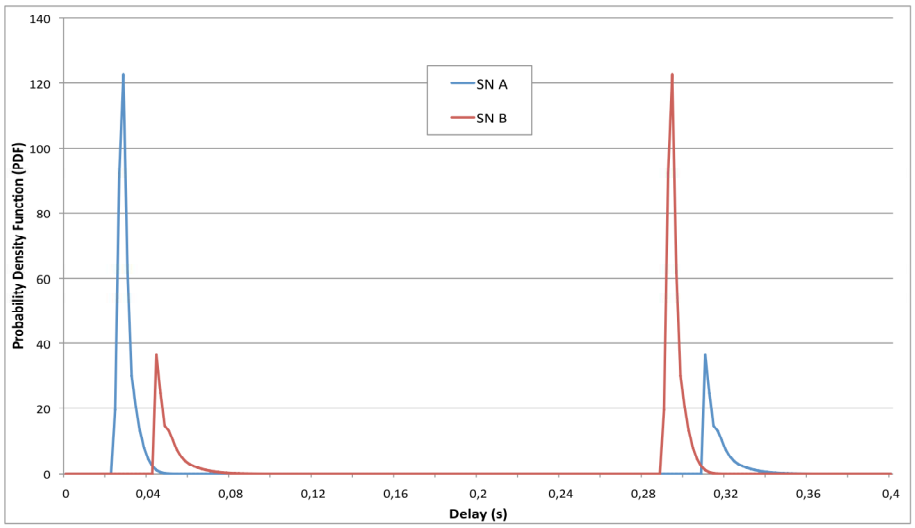


Figure 4.10: PDF of the delay. Scenario 2

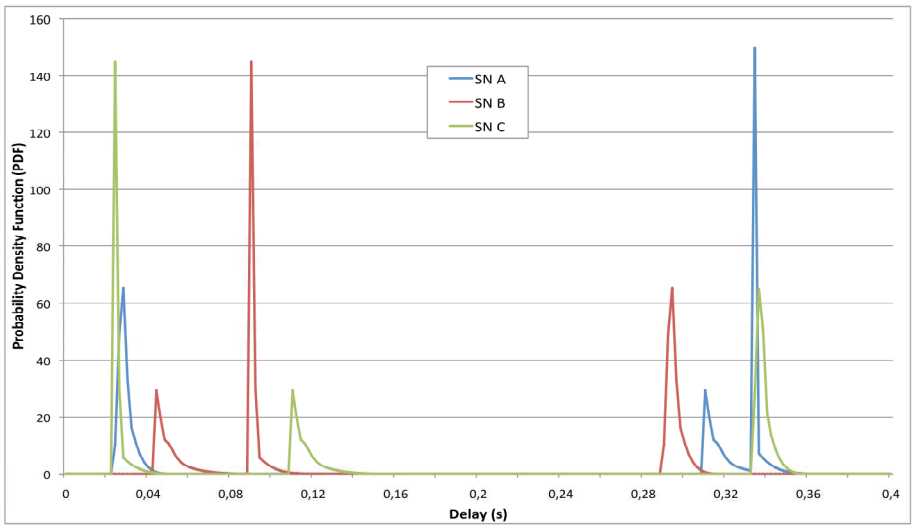


Figure 4.11: PDF of the delay. Scenario 3

RESTRICTED DOCUMENT

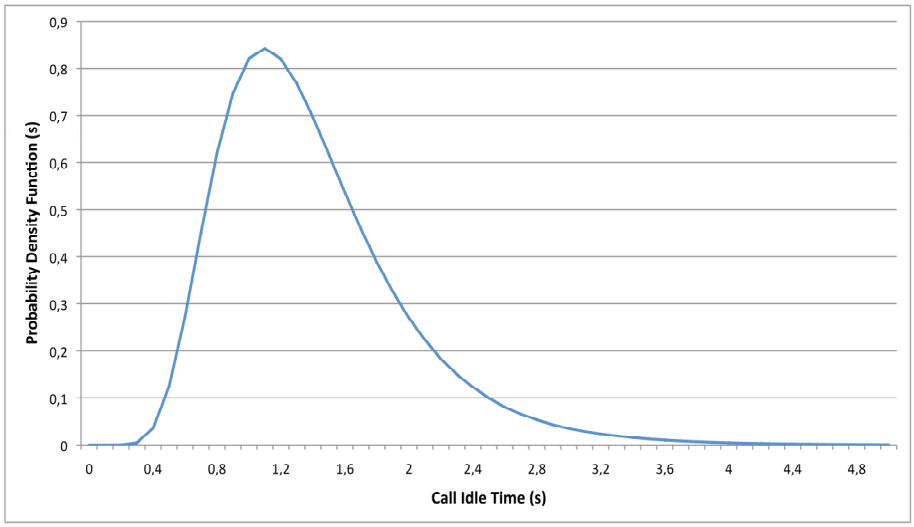


Figure 4.12: PDF of the Call Idle Time

Table 4.1: Simulation parameters

Call Idle Time - Lognormal		
$\mu = 0.25$ $\sigma = 0.4$		
Delay Weibull Mixture		
SN A		
$q_1 = 0.475$	$q_2 = 0.325$	$q_3 = 0.2$
$r_1 = 0.005$	$r_2 = 0.011$	$r_3 = 0.012$
$s_1 = 3.2$	$s_2 = 1.9$	$s_3 = 2.2$
SN B		
$q_1 = 0.45$	$q_2 = 0.3$	$q_3 = 0.25$
$r_1 = 0.012$	$r_2 = 0.002$	$r_3 = 0.007$
$s_1 = 1.3$	$s_2 = 2.25$	$s_3 = 2.1$
SN C		
$q_1 = 0.55$	$q_2 = 0.275$	$q_3 = 0.175$
$r_1 = 0.002$	$r_2 = 0.001$	$r_3 = 0.007$
$s_1 = 3.2$	$s_2 = 1.7$	$s_3 = 1.2$

RESTRICTED DOCUMENT

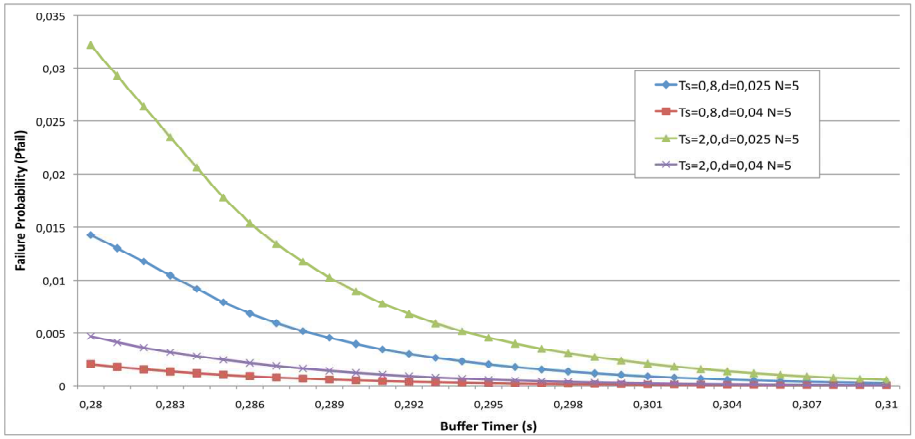


Figure 4.13: Probability failure vs T_{buffer} (large values). Scenario 1

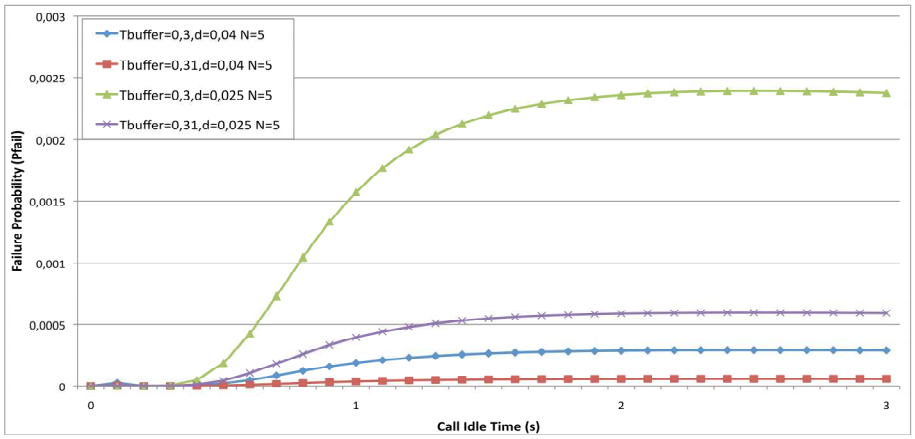


Figure 4.14: Probability failure vs Call Idle Time. Scenario 1

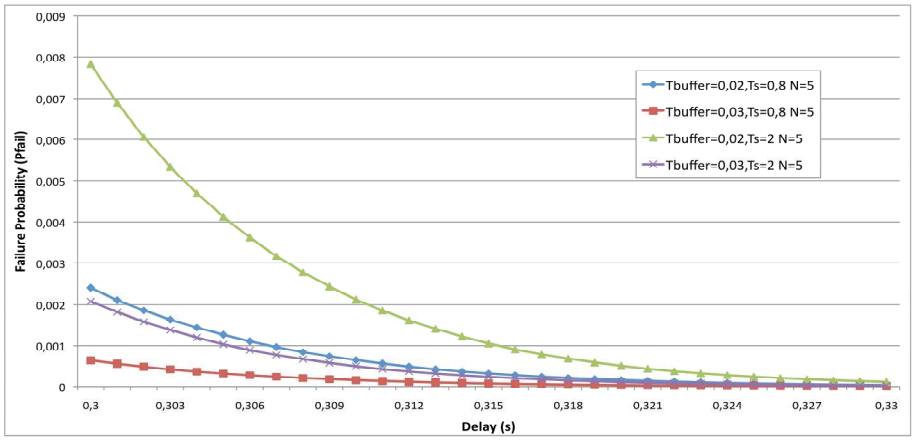


Figure 4.15: Probability failure vs Delay (large values). Scenario 1

RESTRICTED DOCUMENT

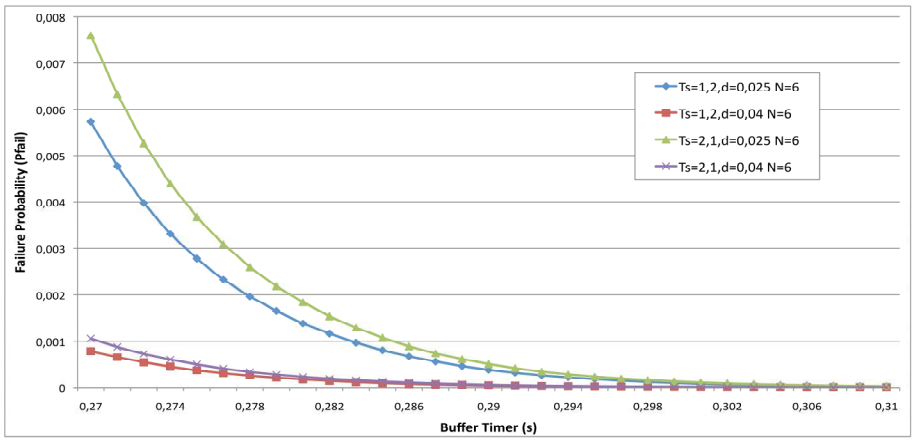


Figure 4.16: Probability failure vs T_{buffer} (large values). Scenario 2

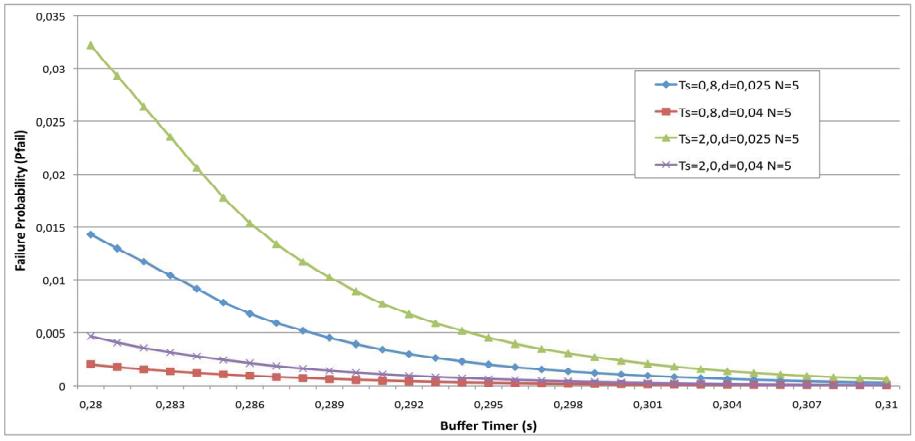


Figure 4.17: Probability failure vs T_{buffer} (large values). Scenario 3

a request coming from the SN B is smaller, as we know the share of requests coming from there. In a different manner, we observe that the $P_{failure}$ is higher for the third scenario because the share of messages coming from SN B and C is greater, so it is likely that a request from the other side of the satellite arrives. Additionally, one should notice that the requests coming from SN C have a larger minimum delay, therefore, the buffer also needs to increase in order to give time to these to arrive.

This shows that the protocol is flexible. Given the same target for the $P_{failure}$, in SN A, the buffer timer duration will be smaller for the requests coming from its users in scenario 2 compared to scenario 1.

4.8.2 Simulation and characterization of the parameters

In this second part of the evaluation, the event driven ns-3 simulator [83] was used to imitate the behavior of a real network with several users issuing requests as they would do in the three scenarios. We wrote the client and server part of a distributed PMR solution that employs the described protocol.

RESTRICTED DOCUMENT

The client part works as a generator of requests and waits for the reaction of the servers. Each request has a header with the value of the timestamp and the sequence number. Call idle times are randomly generated following a log-normal distribution. Once a client receives the pre-grant message, it sends a set of packets that imitates the voice messages and a floor release at the end.

The server node has a block that estimates the parameters of the call idle time and the system delay. The values computed by the parameter estimation blocks from the SNs may differ from the values imposed at the user level but, as the distributions are flexible enough, they provide with a set of parameters that has a good fit and serves to calculate the failure probability and the necessary timers. Further, it computes a set of T_{buffer} on a pre-defined frequency. It simulates the buffering of the voice packets that are then sent to the clients upon the expiration of the buffer timer. Each server works independently and the calculated matrix T_{buffer} varies slightly. Finally, every SN counts the number of failures occurred during the simulation and displays the corresponding $P_{failure}$ result.

In these simulations, clients are connected to their SNs via point-to-point links with a delay similar to the one experienced in LTE networks (20 - 50 ms). An additional random delay is introduced, with a generator based on a mixture of two-parameters Weibull distributions, to emulate the possible buffering and queueing delays. Furthermore, SNs were interconnected with links experiencing higher delays as showed in figures 4.7 and 4.8.

The protocol was conceived to be fair among all users, therefore the call idle time is a time relative parameter generally provided by the user. However, the system cannot assure that all users receive the release at the same time and therefore the beginning of the call idle period for each user does not start at the same time. The release message is buffered for the same period of time of the call, which usually helps to reduce the differences. Yet, some extra errors occur because of this mismatch between release arrivals at the user level.

In some Super Nodes and in some scenarios, it may be necessary to increase the buffering period for the release message when a local user has talked. We have introduced an extra wait to compensate this issue but it is expected that it will still occur from time to time. For this, we have recovered the results of the system errors considering these cases and the ones generated by the model itself. We call the latter protocol failures and the target is for these type of errors. We have performed a series of simulations with the same scenario and different $P_{failure}$ targets in order to see if the resulting $P_{failure}$ matched the given objective. Table 4.2 shows the obtained results:

For the first target of 10^{-2}, we observe satisfactory results for the nominal case, counting all errors, and when considering only the ones caused specifically by the protocol operation itself. These values are inferior to the actual target, mainly because in many cases a T_{buffer} of 0 was already providing a $P_{failure}$ smaller than the target.

Considering the target of 10^{-3}, we see that the delay compensation we introduced was not able to compensate enough and the nominal rate was larger than the target. This is particularly clear in the third scenario, where there were three Super Nodes and the compensation was more complicated. Results were again under the target for the protocol failures.

RESTRICTED DOCUMENT

Table 4.2: Simulation results

	Pfailure Nominal			Pfailure Protocol		
Target	10^{-2}	10^{-3}	10^{-4}	10^{-2}	10^{-3}	10^{-4}
Scenario 1	$3,59*10^{-3}$	$1,63*10^{-3}$	$1,96*10^{-3}$	$1,25*10^{-3}$	$4,36*10^{-4}$	$2,51*10^{-4}$
Scenario 2	$4,21*10^{-3}$	$1,69*10^{-3}$	$1,16*10^{-3}$	$1,46*10^{-3}$	$7,52*10^{-4}$	$2,54*10^{-4}$
Scenario 3	$6,93*10^{-3}$	$4,03*10^{-3}$	$4,79*10^{-3}$	$1,15*10^{-3}$	$3,36*10^{-4}$	$1,32*10^{-4}$
Average	$6,18*10^{-3}$	$2,08*10^{-3}$	$2,00*10^{-3}$	$1,29*10^{-3}$	$5,08*10^{-4}$	$2,12*10^{-4}$

Finally, the outcomes of the most constrained target, 10^{-4}, were not that positive. We observe, for the nominal case, results were close to the ones obtained for the previous target. Indeed, we observe an increase in two scenarios. We believe this is a result of the increase of buffer lengths actually. The delay compensation we introduced was fixed for all target, therefore, if the buffer was even larger, this compensation was not enough. We see that the results for the protocol failures were higher than the target. We think that this is given in part by the inaccuracy of the simulation. First, the minimum unit was milliseconds and second, most of the values were obtained by applying interpolation from a fixed set of values, calculating the actual T_{buffer} in steps of 5 ms for the delay. Recall that as no closed-form solution exists for the calculated probabilities, the use of numerical methods could result in a decrease of the precision of the procedure.

In general, we are satisfied with the results. We did not intend to have a perfect delay compensation because we expect to have this type of inaccuracies in real scenarios. We think that the default target for the $P_{failure}$ should be 10^{-3}. More restricted target would not be met if there not exist a very good compensation in some cases.

Annex: Computation of the failure probability and buffer timer calculation

This annex presents the integrals of the distribution probability functions that define the probabilities in equation (4.4).

These integrals were calculated using numerical methods such as the adaptive Simpson's rule.

$$\begin{aligned} &Prob\left\{T_1+d_1+T_{buffer}<T_2+d_2 \quad \& \quad T_2<T_1\right\}= \\ &Prob\left\{T_1+d_1+T_{buffer}-\tau<\delta \quad \& \quad \tau<T_1\right\}= \\ &\int_0^{T_1}\int_{T_1+d_1+T_{buffer}-t}^{+\infty} f_\tau(t)f_\delta(d)d\delta d\tau \end{aligned} \tag{4.11}$$

$$\begin{aligned} &Prob\left\{T_1+d_1<T_2+d_2 \quad \& \quad T_2<T_1\right\}= \\ &Prob\left\{T_1+d_1-\tau<\delta \quad \& \quad \tau<T_1\right\}= \\ &\int_0^{T_1}\int_{T_1+d_1-t}^{+\infty} f_\tau(t)f_\delta(d)d\delta d\tau \end{aligned} \tag{4.12}$$

RESTRICTED DOCUMENT

$$
\begin{aligned}
&Prob\{T_1 + d_1 < T_2 + d_2 \quad \& \quad T_2 > T_1\} = \\
&Prob\{T_1 + d_1 - \tau < \delta \quad \& \quad \tau > T_1\} = \\
&\int_{T_1}^{+\infty} \int_{T_1 + d_1 - t}^{+\infty} f_\tau(t) f_\delta(d) d\delta d\tau
\end{aligned}
\tag{4.13}
$$

RESTRICTED DOCUMENT

Chapter 5

Satellite Access Strategies

The efficient use of the scarce satellite resources is a well-studied subject. From the beginning of satellite networks the solutions have evolved but the fundamental strategies have remained similar. We evaluate them in this chapter and we detail how they are used in DVB-RCS/2. Then, we discuss the best strategies to adopt in order to provide a good quality of service to the push-to-talk application and maintain an efficient use of the satellite bandwidth. Later, we present a model to estimate the packet inter arrival process of a PTT conversation. Finally, we perform an experiment comparing several resource demand options to manage PTT traffic.

5.1 Introduction

In a scenario such as the one considered in this thesis, a temporary LTE cellular network backhauled through a satellite link, the satellite terminal could use a return-link based standard. Instead of having a fixed allocated bandwidth, a satellite terminal with low symbol rate uses a bandwidth that is shared among multiple stations. Therefore, an efficient use of the resources may result in an increase in terms of number of users and better overall performance. Basically, there exists three ways to access the return link in this competing scenario: fixed access, demand access and random access.

Fixed-Assignment Multiple Access (FAMA)

The users receive a fixed quantity of resources throughout the duration of their connection. The resources can be assigned using a frequency division scheme (FDMA), a time division plan (TDMA) or an access through orthogonal codes (CDMA). While this option may work well for users that have a fixed transmission pattern, it is very inefficient when the stations do not have any data to transmit. At the same time, there is not a way to increase the sending rate momentarily to benefit from the resources the other users do not need.

Demand-Assignment Multiple Access (DAMA)

DAMA is a more efficient resource allocation technique where the channels are assigned upon the demand of the multiple users. The allocation is done in real-time, reducing the unused portion of the spectrum and improving the performance given by FAMA systems. However, the delay between the demand request to the resource manager and the approval may be an issue especially for real-time services such as voice or video transmission.

In the case of satellite networks there are two approaches: the demand assignment process can be handled by a terrestrial large station or by an entity at the satellite itself. In the first case, the satellite is transparent and the minimum delay corresponds to two propagation hops through the satellite link. In the second case, the satellite features an on-board processor (OBP) allowing reducing in a half the minimum allocation delay. [84] provides an overview of the capabilities of an OBP, in this case the one developed by the German Fraunhofer Institute.

Generally, capacity is allocated frame-by-frame and the stations have a specific resource, a signaling channel, in time or frequency, to perform their capacity requests. The resources demands have to be renewed frequently.

DAMA has been deeply studied in the literature and there are multiple works that propose methods to increase performance and reduce the resources needed to send the capacity requests. For example, authors in [85] suggest burst targeted DAMA (BT-DAMA) designed for handling traffic that comes in bursts in an ON/OFF pattern. They propose that the terminals request the permission to send as usual but instead of renewing their request, they only inform the allocation system when their transmission ends. Another solution allows the users that are already transmitting to send their capacity requests along with their data. This technique is also known as piggy-backing [86]. A procedure analogous to BT-DAMA was proposed in [87] to handle group communications over satellite, a service similar to the PTT applications.

Random Access (RA)

Satellite random access is a good alternative to the previous discussed approaches and it may be preferred in case the data transmitted is bursty, i.e. short and intermittent. RA requires no central entity, at the satellite or on the ground, to control the user access. However, collisions of bursts sent by different users may occur and require, then, the retransmission of those frames, increasing the overall end-to-end delay. Given the nature of satellite links, it is impossible to use carrier sensing and collision avoidance mechanisms, which are common in other wireless channels. The inherent delay of geostationary links prevents the terminals to have an immediate channel sensing.

Aloha-based techniques [88] have been used in satellite networks for more than 40 years. In pure Aloha, the station transmits a frame without prior channel sensing. If the station receives data from other station during the transmission or immediately after it, there has been a collision and the data will have to be reissued later after a random back-off time.

RESTRICTED DOCUMENT

The slotted version (SA) improves the efficiency of the pure protocol. Time is divided in discrete timeslots and stations can only send at the beginning of a timeslot, reducing the number of collisions. Figure 5.1 illustrates the difference between the two versions and shaded frames represent collisions. SA has shown to be a good option under low congestion scenarios.

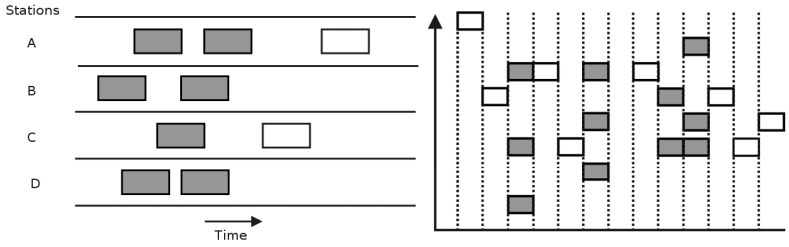


Figure 5.1: Pure and Slotted Aloha

Starting from the Aloha protocols, improved versions have been introduced in the last decades. For example, [89] introduces an enhancement called Diversity Slotted Aloha (DSA). A terminal randomly transmits multiple copies of the same packets in different frequencies or separated by random intervals. This improves the probability that at least one copy is received without collision. Yet, the performance decreases considerably once maximum throughput is achieved due to the multiple copies that increase the total channel load.

Another possibility is to use spread spectrum techniques. In [90] authors propose Spread Spectrum Aloha (SSA). A terminal selects a random spreading sequence among a predetermined set and uses it to transmit the data packet. As long as the terminals that have transmitted in the same slot use different sequences, the receiver will be able to recover the different frames. Nevertheless, a high number of sequences increases the receiver complexity. In SSA the throughput grows linearly with the channel load until a breakdown point is reached, when the signal-to-noise plus interference ratio (SNIR) is above the accepted limit.

Finally, a more recent mechanism has been presented in [91]. Contention Resolution Diversity Slotted Aloha (CRDSA) takes the approach of DSA and transmits two copies of the same packet separated by a random time. Efficiency is improved by using interference cancellation techniques. The copies contain a pointer to the location of the other copies. These pointers are used in order to attempt restoring collided packets by subtracting the interfering content of already decoded packets. If at least one copy of a given packet has been correctly received, its interference from other copies of the same packet can be removed thanks to the knowledge of their location.

5.2 Access control in DVB-RCS/2

DVB-RCS is the specification from the Digital Video Broadcasting Forum that defines a Return Channel via Satellite. It provides interactive multimedia communications for small satellite terminals. The second

RESTRICTED DOCUMENT

version, DVB-RCS2, introduces QoS management, adaptive coding and modulation (ACM) and the use of random access.

Figure 5.2 shows the general architecture of such systems. Two channels, a forward or broadcast channel and a return or interactive channel compose the satellite network. Stations are referred as Return Channel Satellite Terminals (RCST). The Network Control Center (NCC) is the central control entity that manages the resource allocation mechanism, which could also be located on board of the satellite. Finally, the gateway (GW) redistributes the received packets to the external networks.

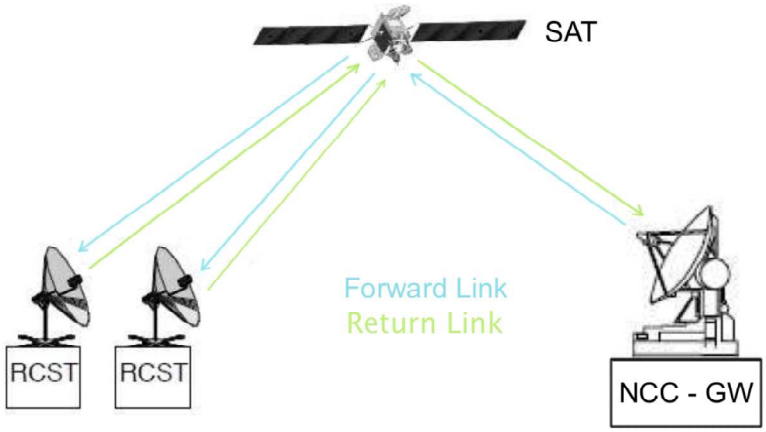


Figure 5.2: DVB-RCS reference scenario

In DVB-RCS systems, there is a specific part of the spectrum devoted to access through DAMA techniques, the dedicated access allocation channel (DA-AC). The RCSTs send capacity requests (CR) to the NCC, which responds with the allocated resources to each user. There exist different types of capacity requests depending on the QoS expectations of the data that needs to be sent. The capacity categories defined in the DVB-RCS framework are:

- *Constant Rate Assignment (CRA):* Rate capacity provided at full for each terminal while logged on the system. Generally, this capacity is not requested and depends on the subscription parameters of each user. It reminds the fixed assignment (FAMA) scheme, but it is more flexible as it allows redistributing the resources in case the terminal does not transmit.

- *Rate Based Dynamic Capacity (RBDC):* Rate capacity that is requested dynamically. The request has to be renewing considering the current queue occupancy.

- *Volume Based Dynamic Capacity (VBDC):* Volume capacity that is requested dynamically. Requests can be cumulative or demand an absolute volume (AVBDC).

- *Free Capacity Assignment (FCA):* The NCC has the possibility to allocate the unrequested resources among the users to momentarily boost their transmission capabilities.

RESTRICTED DOCUMENT

DVB-RCS2 generally uses the Differentiated Services (DiffServ) framework to provide QoS management. Input traffic is classified and each class defines a request class (RC), which describes a method to provide resource allocation in the dedicated access channel. RCs can accommodate different capacity categories defined above. DiffServ classifies traffic in three main groups:

- *Expedited Forwarding (EF):* Accounts for real-time traffic that requires low loss, low latency and low jitter. As it is considered as the premium class and tends to vary slowly over time, CRA is a suitable option to allocate this class. RBDC may be supplement in case the maximum CRA quota has been achieved. However, a higher delay will be introduced for the request and response process.
- *Assured Forwarding (AF):* It defines the critical data category that needs guaranteed delivery but that tolerates a higher delay. RBDC is suitable as it offers a capacity guarantee and VBDC may be a complement for lower priority traffic.
- *Best Effort (BE):* It covers the remaining traffic that does not have specific QoS requirements. The VBDC category handles this low priority class.

DVB-RCS2 also offers the opportunity to use a Random Access Allocation Channel (RA-AC), exploiting the previously described CRDSA protocol.

RCSTs perform traffic classification and decide which is the best strategy in order to define the request classes. The allocation mechanism in the NCC is usually designed to maximize the overall utilization of the return channel. RCSTs may jointly use DA-AC and RA-AC and reallocate the frames between the two methods to effectively handle the incoming traffic.

5.3 Allocation strategies to handle PTT Traffic

Push-to-Talk traffic comprises two categories of traffic: signaling and voice frames. Signaling transmitted through the satellite link includes floor requests and releases. Additionally, floor denies and grants could also be transmitted while it is not necessary given the described floor control protocol. Voice frames have a high priority level, similar to the one of regular voice calls.

Signaling frames are short packets that require low delay and guaranteed delivery. It is preferred that they are handled by CRA if possible. Alternatively, the random access channel could also be used in case the congestion over it is limited and success probability stays high.

Given this approach, the inherent delay of the dedicated assignment could be avoided and the core network would be rapidly informed of the incoming requests.

PTT voice packets are of a different nature. They consist of encoded voice frames that have a constant generation interval of a few tens of milliseconds, generally 20 ms. Calls have usually a short duration and there is a period of no transmission between different calls. Multiple calls form a conversation, which can be recognized because the average inter conversation time is clearly larger than the call idle time.

RESTRICTED DOCUMENT

Voice packets are still small compared to other services and they usually consist of a IP/UDP/RTP header of 40B and a payload of 20B-60B depending on the encoding protocol.

Using CRA to accommodate PTT voice would be perfect from the QoS point of view. Yet, PTT traffic is shorter and less constant compared to other VoIP services. Therefore, CRA could be overestimated and the resources lost in case the station handled only PTT traffic.

In case it is not possible to use CRA, it is important that the request delay and the initial call delay are similar. If not, situations like the ones shown by figure 5.3 occur. In this case a floor request is sent via CRA resources or using the random access channel. On the other hand, a capacity request is issued to handle the voice frames, considering a RBDC request. The overall delay makes the buffering mechanism introduced in the floor control protocol useless.

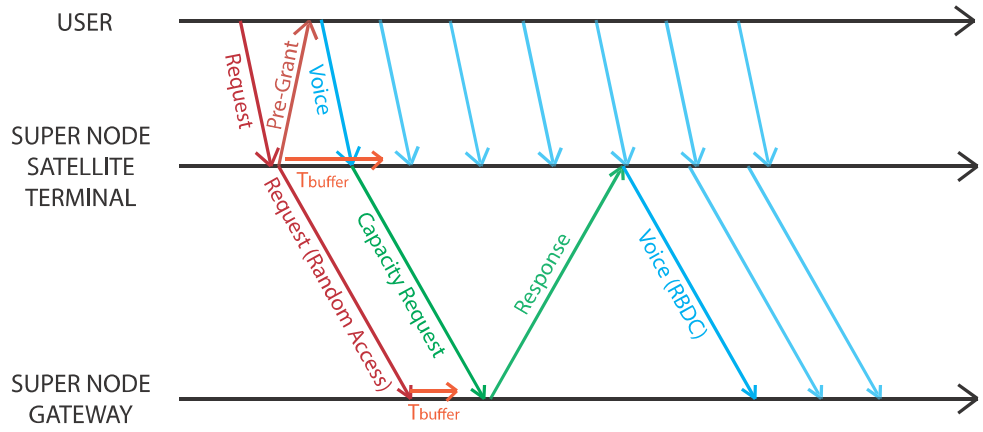


Figure 5.3: Delay difference between signaling and voice data

In order to avoid this scenario, the best is to exploit the random access channel to send the first voice packets. As the transmission is expected to last a few seconds, the RA channel could be used throughout the call in case the load of this channel remains low. However, to increase the robustness, it is preferable to send a CR along with the first voice frames in order to obtain dedicated resources. Therefore, the first packets will be sent through RA until the allocation confirmation is received. A rate based CR would be suitable for handling PTT voice until its end. The CR could also be sent at the same time of the floor request, even if the probability that another user gets the floor is still high. If the final holder is another user managed for the return station, the resources intended for the first user would be given to the final floor holder. Figure 5.4 provides a visual overview of this approach.

Another possibility would be to use prioritization and use already allocated resources, requested for low priority or best effort traffic that consider a volume-based category. In this case, a VBDC update should be sent along with the RBDC request for the PTT voice.

An additional issue of the DAMA procedure is that when transmission is over, the RBDC request needs to be updated in order to inform the NCC. The satellite terminal informs the NCC of the end of the call after the transmission of the last voice packet. However, there would be an overallocation until the allocation update occurs.

RESTRICTED DOCUMENT

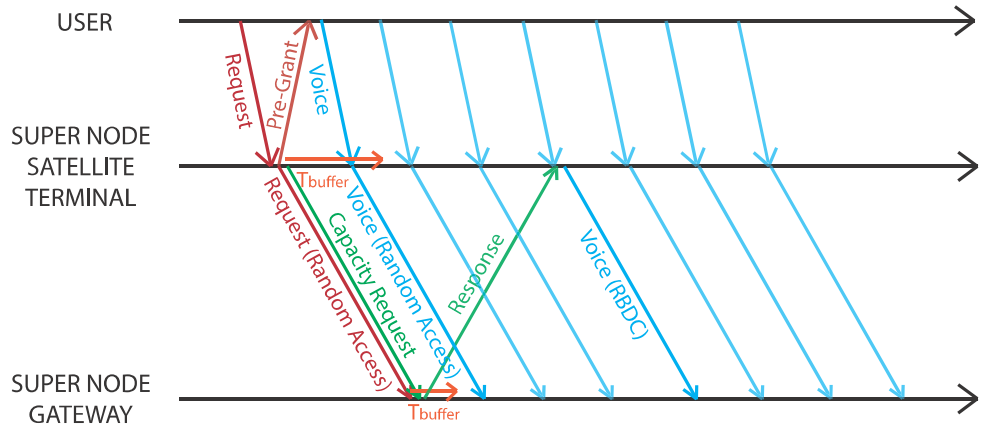


Figure 5.4: Proposed solution

It is possible to use predictive methods to estimate the end of a call and send the RBDC request. Then, the last voice frames will be sent through the RA channel. Yet, the call could end before the predicted end or largely after, so it should not be the preferred solution.

Predictive allocation could also be used between calls. In case the NCC could be optimized, predictive allocation could influence the FCA allocation. The NCC may detect the ongoing PTT conversations and assign free resources to stations during the call idle times. These could be use to send floor requests, capacity requests and the first voice packets.

Given that call idle times are monitored by the floor control protocol, they might be use for predictive allocation. Therefore, FCA would only be assigned during the intervals where most of the floor request and consequent voice packets are sent.

Additionally, in groups where conversations usually imply a call from one side of the satellite link followed by another from the opposite side, FCA could not reserve resources for a group when it is probable that the next call will come from the core network.

Perfect predictability is not possible and it should only be used in case it may improve the overall resource efficiency. However, predictive allocation would increase the complexity of the allocation mechanism and would probably result in over resource allocation to the stations with ongoing conversations. Therefore, we reaffirm that random access cold start should be the preferred strategy followed by rate reservation requests. Predictive allocation could only be useful in systems with no RA channel (such as DVB-RCS) and with low congestion and, consequently, sufficient FCA resources available.

5.4 PTT Traffic Conversation Model

In the previous discussion, we have stated our hypothesis about the best methods to manage the on-demand resources for push-to-talk transmissions. The rest of the chapter is dedicated to examine the validity of our conclusions.

Our interest is in the transmission of multiple PTT conversations, a situation we would probably

RESTRICTED DOCUMENT

encounter in disaster management after we deploy a temporary mobile network. Conversations include calls from both sides of the satellite link as well as conversations between users in the deployed area that are retransmitted to the core network for monitoring purposes.

Predictive resource allocation for a single PTT conversation is hard as we just stated. Predicting when the call is going to end or start is challenging. However, in case there exists multiple ongoing PTT conversations, is there a way to estimate the total necessary results? To answer this question we have first to observe if we can model a PTT conversation and then, try to move towards a model superposing multiple PTT conversations.

5.4.1 VoIP ON-OFF Model

First, we review the model of a voice over IP call. Traditionally, VoIP has been characterized by a ON-OFF model: A call can be in two states, either in ON or in OFF. The codec features a voice activity sensor to stop the packet transmission during silence, which refer to the OFF state. When the user is speaking, we find the call in the ON state, sending a packet each T milliseconds, given by the inter arrival time of the codec, usually around $20-30$ milliseconds. This model for artificial conversation speech was adopted by the ITU-T in 1993 [92]. We observe this model in figure 5.5.

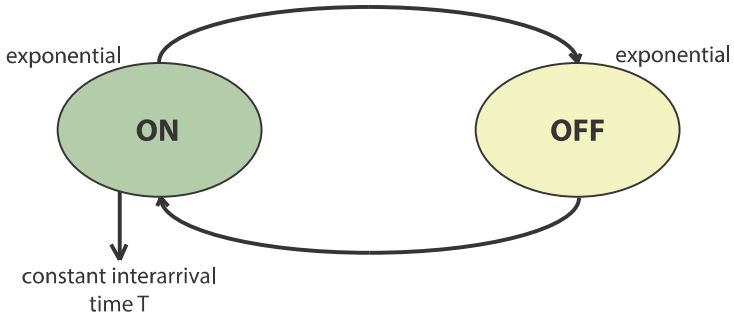


Figure 5.5: VoIP ON-OFF Model

The duration in both states are characterized by a exponential distribution with mean T_{ON} for the ON state and T_{OFF} for the OFF state. The classical values for the means are $T_{ON} = 352$ and $T_{OFF} = 650$, in milliseconds.

In addition to this model, the arrival of new calls have usually been modeled by a Poisson distribution with an exponentially distributed time between new call arrivals.

5.4.2 PTT Conversation Model

Several works have been devoted to the study of Public Safety traffic on legacy PMR networks [93] [94] [95] [96] [97] [80]. Push-To-Talk is a half-duplex communication mean. Each call lasts a few seconds and two or more users exchange multiple calls with a small idle time between them. A series of calls can be regarded as a conversation. Following this logic, [80] derived the three-state model shown in figure

RESTRICTED DOCUMENT

5.6. A given talk-group is in call holding state when any of its users is speaking. After a short call is finished, the model can move to the call idle state until some other users answers and forces to change back to the call holding state. Once a conversation ends, the model transitions to the conversation idle state. Generally, a threshold is used to determine if the conversation has finished and the group is in this state. Authors decided to use 3 seconds as threshold value.

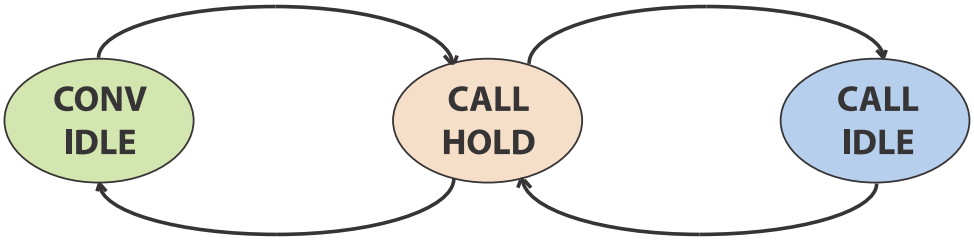


Figure 5.6: PTT Three-states Traffic Model

This model is a simplified version of a n-state model that included the probability of a conversation to have a determined number of calls. First studies from other authors referenced at the beginning of this subsection already stated that the duration of each of the commented states was unlikely to be exponentially distributed. From the analysis of the traffic from the Public Safety network of Bonn, Germany; the authors concluded that the random variable that best fits the three states is the lognormal distribution, with different parameters for each case. Weibull and gamma distributions also offered a good fit for some of the states.

The estimation of the PTT traffic in a given network is challenging. When examining the traffic traces from a long period of time, the authors observed a long-range dependence of inter call arrivals, i.e. correlation between distant events in time. This phenomenon is also known as self-similarity.

They also observed that the differences between traffic intensity over time came from the nature of the work of the users. They divided the data between the traffic from regular first responder activities and the one observed in disaster areas. The latter case meant a heavier traffic for the network, specially observed in the reduction of the length of the conversation idle periods, meaning that there were more conversations. Yet, compared to the first responder regular traffic, call holding times were slightly shorter and conversations usually had less calls.

With these conclusions, we see that estimating the traffic generated by each talk group depends largely on the type of group and on the activity they are performing at a given instant.

Nevertheless, we observe that the distribution of call holding times and call idle times did not differ much when comparing first responder and disaster area traffic. With this in mind, we proceed with the analysis of the traffic generated from a given conversation. Instead of estimating the arrival of new conversation, the resource demand mechanism has in interest the characterization of a conversation, to reserve enough resources during its duration.

We analyze the traffic generated by a PTT conversation observing the inter arrival time of voice packets. Similar to the VoIP source model, we expect to have voice activity detection activated at the

RESTRICTED DOCUMENT

codec. While we do not have any data supporting it, we also expect that the duration of the silence periods, within a call, will be significantly smaller compared to the VoIP model. Therefore, a PTT conversation is shown in figure 5.7. A conversation has multiple calls, separated from a given call idle period. Each call has itself ON and OFF periods, only transmitting packets during ON states.

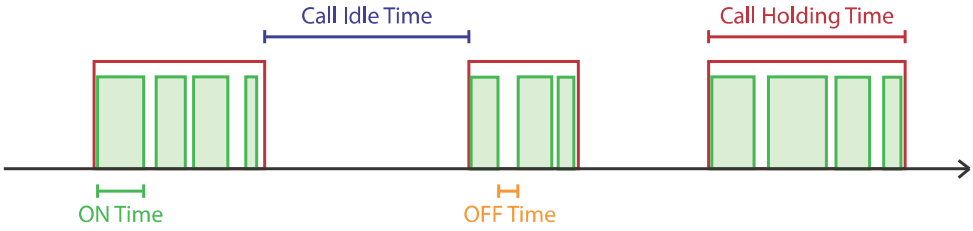


Figure 5.7: PTT Conversation

Hence, we derive our PTT conversation model presented in figure 5.8. There are basically two main states, call hold and call idle. Then, during call hold periods, we observe ON and OFF states. We will later analyze the traffic generated by this model.

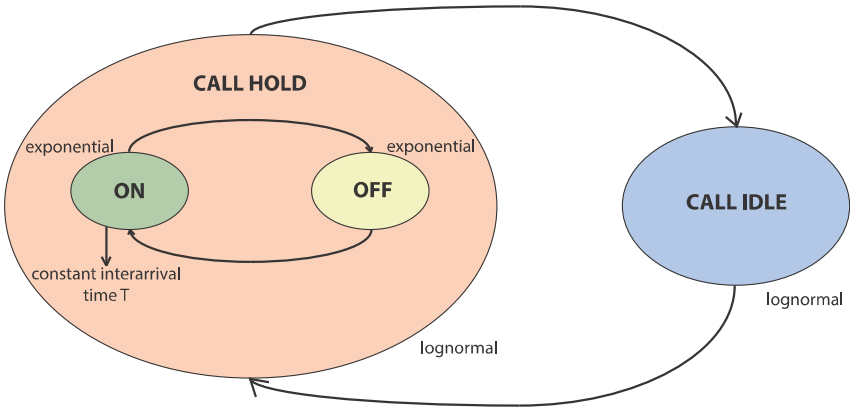


Figure 5.8: PTT Conversation Model

5.4.3 Superposition of multiple PTT Conversations

[98] analyzed the aggregate packet arrival process of the superposition of separate voice streams starting from the VoIP source model presented previously. VoIP model was seen as a renewal process. However, the packet inter arrival distribution from this process is highly variable, we observe that most of the inter arrival times are just one packetization period, T, usually 20 milliseconds; but occasionally, after silence periods, this inter arrival is much longer.

The aggregate inter arrival process is very bursty and the instantaneous arrival rate depends on the number of active sources, which tends to fluctuate substantially. The authors of the study focus on the dependence among successive inter arrival times in the aggregate process. For this purpose, they observe the index of dispersion for intervals (IDI). In [98], the IDI is defined as follows:

RESTRICTED DOCUMENT

Table 5.1: Superposition model parameters

ON Period		Call Holding Time (seconds)		
exponential	$mean = 0.350s$	lognormal	$\mu = 0.89974$	$\sigma = 0.68727$
OFF Period		Call Idle Time (seconds)		
exponential	$mean = 0.200s$	lognormal	$\mu = -0.34817$	$\sigma = 0.84829$

Let $\{X_k, k \geq 1\}$ represent the sequence of inter arrival times of a process and let $S_k = X_1 + ... + X_k$ denote the sum of k consecutive inter arrival times. The IDI, called the k-interval squared coefficient of variation sequence, is the sequence $\{c_k^2, k \geq 1\}$ defined by:

$$c_k^2 = \frac{kVar(S_k)}{[E(S_k)]^2} = \frac{Var(S_k)}{k[E(S_k)]^2} \tag{5.1}$$

We are going to analyze first the probability density function of the packet inter arrival time and compute the k-interval squared coefficient of variation sequence of the superposition of multiple PTT conversations. Our simulations were performed with MATLAB™. The parameters used are listed in table 5.1.

Call holding and idle time parameters have been extracted from [80]. Considering the ON-OFF durations, we have taken those of the classical model for the ON periods but we have decided to adopt shorter OFF periods, as we discussed previously. All parameters are in seconds.

We compare the probability density function of the packet inter arrival time from a single PTT conversation source with the one of a VoIP source with the same ON-OFF parameters in figure 5.9. Both PDF have a large delta at $T = 0.02$ seconds, so we focus on the evolution of the PDF after this delta, i.e. $t > 0.02$.

We observe that the shape of the PDF for the inter arrival time of one PTT source and one VoIP are similar. The PDF for one PTT source fluctuates around the exponential decay of the OFF period duration because there are more silence periods than call idle ones. However, the mean of the distribution is larger for PTT source as idle times are usually larger than the silence periods. We see that the larger the mean of the OFF state, the smaller the difference between the mean of the PTT and VoIP source models, which is normal as the mean duration of the silence periods approaches the mean of the call idle time.

Next we present our results for the superposition of multiple PTT conversations in table 5.2. We derive the mean and the variance of the packet inter arrival time (IAT) of the aggregation process. We also computed the division between the mean of one source by the number of sources and we observed that it approaches the estimated mean. Finally, we include the mean of the one VoIP source for comparison.

We conclude this subsection with the analysis of the index of dispersion for intervals derived from the superposition of multiple PTT conversations, presented in figure 5.10. Our results are similar to the ones obtained in [98] for the superposition of VoIP sources.

Over short intervals, the superposition looks like a Poisson process but over longer intervals of time the superposition significantly deviates from a Poisson process and becomes very variable. Even with the

RESTRICTED DOCUMENT

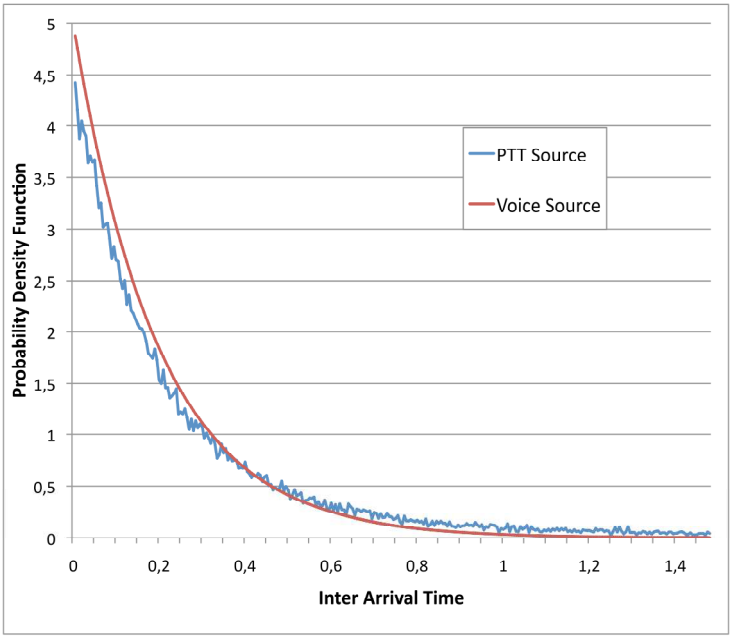


Figure 5.9: Probability Density Function of one VoIP or PTT source

Table 5.2: Superposition Process Results

Number of Sources	1	2	10	25	50
Mean IAT	$3.93 * 10^{-2}$	$1.97 * 10^{-2}$	$3.99 * 10^{-3}$	$1.59 * 10^{-3}$	$7.95 * 10^{-4}$
Variance IAT	$2.45 * 10^{-2}$	$2.33 * 10^{-3}$	$2.52 * 10^{-5}$	$5.88 * 10^{-6}$	$8.20 * 10^{-7}$
mean one source / number of sources		$1.96 * 10^{-2}$	$3.93 * 10^{-3}$	$1.57 * 10^{-3}$	$7.86 * 10^{-4}$
VoIP source Mean		$3.14 * 10^{-2}$			

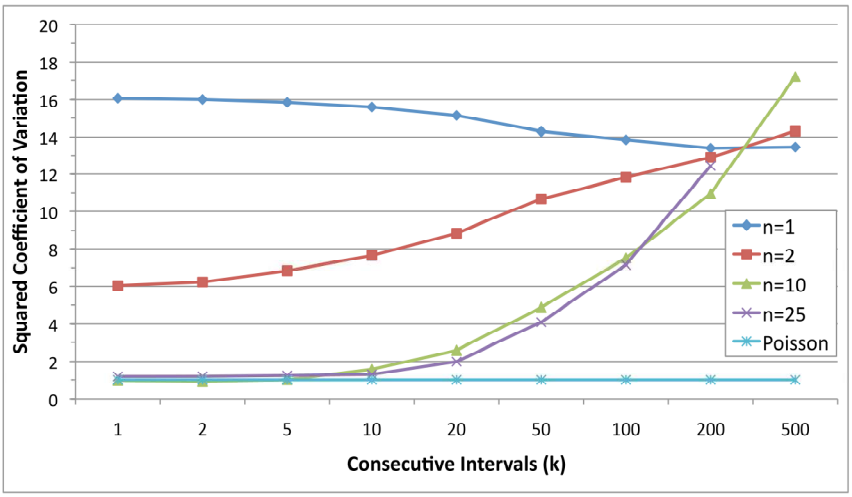


Figure 5.10: Index of Dispersion for Intervals of the Superposition of PTT Conversations

RESTRICTED DOCUMENT

shorter duration of the PTT conversations, the longer multiple conversations coexist, the more disperse becomes its superposition process. Given that the number of conversations managed by a single deployed LTE cell will be limited to around 20 or 30, we do not expect the superposition process to behave as a Poisson process.

5.5 Optimal Resource Demand for PTT Traffic

In the remaining of this chapter we are going to compare several resource demand methods to handle push-to-talk conversations. Recall that a conversation is composed by multiple calls, messages from diverse users. We first discuss about the proposed mechanisms, then we review our simulation model and finally, we present the results of the experiment.

5.5.1 Resource demand mechanisms

For this study we consider five different resource demand methods for comparison. They rely on the categories discussed in the first part of the chapter.

CRA Full Rate A capacity demand is issued at the beginning of the conversation at the voice codec full rate during ON periods. This capacity is maintained throughout the conversation.

CRA PTT Rate A capacity demand is issued at the beginning of the conversation at a rate equal to the average rate of a PTT conversation, the one we estimated in the previous simulations. This capacity is maintained throughout the conversation.

RBDC Full Rate A capacity demand is issued at the beginning of each call at a rate equal to the one used for the voice codec during ON periods. This capacity is maintained until the end of the call, when the release message is received. A new demand is necessary for each call.

RBDC Full Rate + Min Equal to the case *RBDC Full Rate* but with an additional minimum rate that is automatically allocated during all the simulation.

RBDC Full Rate + Random Access Equal to the case *RBDC Full Rate* but taking advantage of the random access mechanism. Nevertheless, we have limited the usage of this mechanism to make it comparable with the previous case but without requiring a formal allocation. Therefore, the maximum rate one can obtain through random access is equivalent to the minimum allocated in the *RBDC Full Rate + Min*.

In all cases, we additionally consider volume capacity demand, VBDC, in case the terminal notices that the queue is building up too quickly. This allows sending the first part of a call quickly in order to allow the receiver to play the message without interruption, similar to the streaming on demand services.

RESTRICTED DOCUMENT

Finally, we consider that the demand rate is updated when the request to send is received at the terminal, before the actual packets start arriving.

5.5.2 Bandwidth on Demand model

The simulation model we have designed for this experiment is based on a PTT traffic generator and a Bandwidth on Demand (BoD) satellite terminal based on DVB-RCS2 [11]. The traffic generator simulates several PTT conversations following the model previously presented. Conversations do not start at the same time, they begin at random moments during the simulation. The generator sends the packets to the satellite terminal as well as the call state information, i.e. the floor requests and releases. The BoD terminal simulates the demand of capacity and sends the packets once it has been allocated some bandwidth. The outcomes we are interested in are efficiency, i.e. the share of the occupied bandwidth compared to the total allocated capacity, and the queuing delay, i.e. the time packets spend on the terminal queue. For this we use a FIFO queue of unlimited size, to observe the effects of the queuing delay. Figure 5.11 provides an overview of this model.

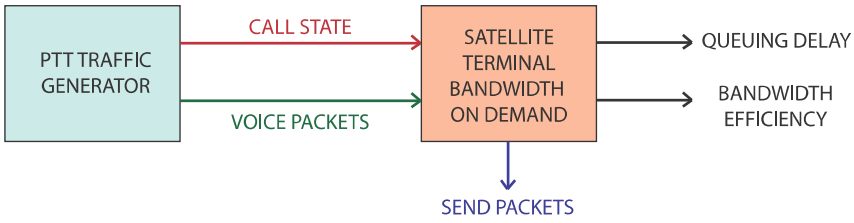


Figure 5.11: Bandwidth on Demand Simulation Model

We try to model the scenario of a deployed LTE cell connected to a DVB-RCS2 satellite terminal. For simplicity the call from all conversations are generated by users in the concerned area and retransmitted to the core network for monitoring purposes. This is the case where the return link of the terminal would have the highest load. We elaborate on what could happen in more complex scenarios, with calls coming from either side of the satellite link, in the results evaluation subsection.

The satellite terminal follows the encapsulation method used in DVB-RCS2, the Return Link Encapsulation (RLE). This is a rather simple method that does not increase overhead excessively. The different encapsulation formats can be observed in figure 5.12.

We will take the simplest encapsulation method, without any optional fields. The standard allows this when using the same higher layer protocols, IPv4 for example. Therefore, the formats we are interested in are the second, third and fourth of the ones shown in figure 5.12. The data field is filled with a higher layer packet, i.e. an IP packet. Un-fragmented packets will use the third format, adding a two-byte header. In case the packet needs to be fragmented, an additional start packet header of two bytes needs to be added. This is observed in the second type, with the Total Length, LT and T fields. The forth format will be use when sending the last fragment of a fragmented packet, adding a sequence number of one byte. We have imposed not fragmenting packets unless there are at least ten bytes available in the burst

RESTRICTED DOCUMENT

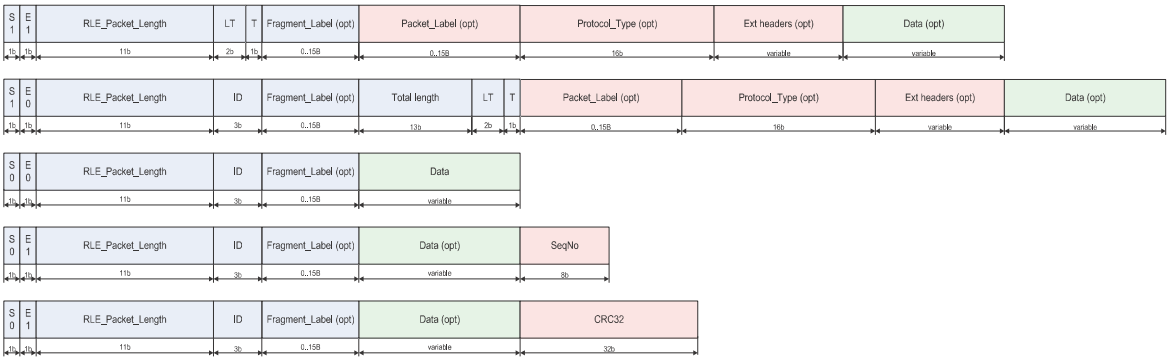


Figure 5.12: RLE Packet Format [11]

payload, in order to avoid excessive fragmentation.

The DVB-RCS2 standard specifies four types of bursts, from very short to very long, each with a different size and transmission duration. In addition, multiple modulation and coding combinations are possible depending on the current carrier to noise ratio. We have decided to use only the short and long types, having a size of 540 and 1620 symbols, respectively. These are the types that allow a higher diversity of waveforms and account for a better tradeoff between efficiency (bits/symbol) and necessary Es/No.

Depending on the waveform used, each burst has a defined capacity in terms of bits available for RLE packets. When measuring the efficiency of a given capacity demand method, we are calculating the ratio of the used bits to the total capacity of the burst. In page 177 of [11] one can find the different type of bursts, waveforms and size of the respective payload.

DVB-RCS2 systems use a Multi Frequency - Time Division Multiple Access (MF-TDMA) to allocate resources among the diverse users. The network management system computes the capacity to allocate to each user and computes a Terminal Burst Time Plan (TBTP), indicating at each frequency channel, which terminal can transmit at each time slot and which type of burst to use. This TBTP is recalculated at a configured rate. The duration of the TBTP allocation is named a SuperFrame. For the sake of simplicity, we consider that each SuperFrame has the duration of roughly the one-way satellite delay. Recall that in order to take into account a capacity demand, the gateway needs first to receive the demand, compute the new TBTP and communicate it to the users. Hence, a total of the satellite round trip delay is necessary before the terminal receives its first allocation. A quick scheme of this procedure is shown in figure 5.13.

We consider that the terminal has its throughput guaranteed because it handles sensitive traffic from an important event, such as an accident. Therefore, as long as its maximum capacity is not exceeded, all the capacity requests will be awarded. This maximum implies the occupation of all time slots of a frequency channel.

The TBTP is computed with the base of allocating the minimum combination of short and long bursts that allocate the user at least the demanded capacity. In order to limit the jitter between consecutive packets, this is done by distributing the burst throughout the TBTP. Additionally, we have included

RESTRICTED DOCUMENT

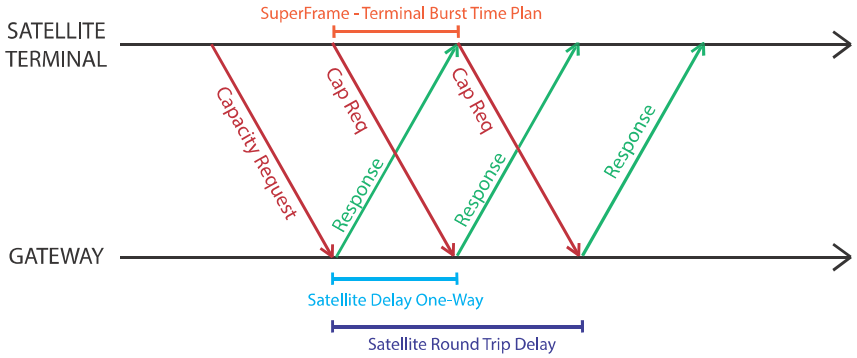


Figure 5.13: Superframe and Capacity Demand

a minimum allocation that allows the terminal to have one transmission opportunity each $15-17$ milliseconds, from the moment it has its first allocation, below the 20 milliseconds mark, which is the inter arrival rate of packets when the conversation is in ON state. This will negatively affect the efficiency in the cases with a very low number of conversations.

5.5.3 Results

The previously presented model has been developed within the ns-3 [83] environment. We simulated different number of groups/sessions. Based on the results of [80], we designed a model for generating PTT conversations. Within a same conversation, the packet arrival rate would be similar to the one we observed above. We used lognormal distributions to characterize the call hold/idle times and the conversation idle time. For the number of calls per conversation, we used a Poisson distribution. We randomly parametrized each group within a range close to the values provided in [80]. Using a uniform random distribution, each group was assigned a value for each parameter depending on the different range.

We wanted to have a model close to a real scenario. Some groups will have more conversations during the simulations, others very few. Even with a large number of groups, say 30 for example, we would never achieve the maximum capacity because the likelihood of having all the groups active is negligeable. We observed that usually the system was managing around 0 to 2 calls and that sometimes we had cases up to 7 - 10 calls. What challenges the system is the arrival of multiple calls in a short period and having to handle them before all the necessary resources are allocated.

The ranges can be observed in Table 5.3. The rest of simulation parameters are provided in Table 5.4. For the voice activity, we used the same values as before: a average OFF time of 200 ms and an average of 350 ms for the ON states.

The combination of these parameters yields a transmission duration of 4 ms for short bursts and 12 ms for the long ones. We performed the experiment with an Es/No ratio of 8 dB. This case would be in between of a full clear sky condition and a severe rain one. For the method adding a minimum capacity, we considered that one call at full rate was allocated throughout the simulation. For all methods, we

RESTRICTED DOCUMENT

Table 5.3: Simulation parameters (all in seconds)

Call Hold Time	min μ	0.7	max μ	1.1
(lognormal)	min σ	0.5	max σ	0.9
Call Idle Time	min μ	-0.55	max μ	-0.15
(lognormal)	min σ	0.7	max σ	1.0
Conversation Idle Time	min μ	2.5	max μ	3.5
(lognormal)	min σ	2.0	max σ	2.5
Call Count (Poisson)	min λ	3	max λ	5

Table 5.4: Rest of simulation parameters

Channel Symbol Rate	135 $ksymb/s$	Satellite One-way delay	270 ms
TBTP Update delay	540 ms	TBTP duration	270 ms
Voice Codec Full Rate	16 kb/s	Inter arrival Time in ON state	20 ms
Voice Packet Size	40 $bytes$	$IP+UDP+RTP$ header size	40 $bytes$
Es/No ratio	8 dB		

performed 20 simulations lasting 10 minutes and we present the average results.

Figure 5.14 presents the results of the efficiency. The first thing we observe is that the efficiency is not very high in general. This has multiple causes. The first is the actual choice of the parameters. The most common case is to have allocated a short DVB-RCS2 frame and have one packet in queue. The frame at $Es/No = 8\ dB$ has 864 payload bits and a single packet has a size of 40 bytes of header, 40 bytes of payload and 2 bytes of RLE header, making a total of 656 bits. Therefore, in that case the efficiency would be $656/864 = 0.76$. Additionally, the application of voice activity detection makes most of the capacity request methods inefficient per se because they demand an allocation at full rate while we know that it will not be used during some periods.

Another cause is the use of *VBDC* allocations. When a call starts, the buffer gets more and more full and we need extra allocation to send the packets. During the call the situation stabilizes and the packets are sent at the same rate they are received. However, when the call ends, the allocation is still active during a satellite round trip delay. This operation reduces the queuing delay at expenses of reducing the efficiency. Evidently, when multiple packets are in queue, fragmentation can be used in order to improve efficiency. Finally, there is an additional cause for this low efficiency. When resources are allocated, we imposed a minimum allocation each $15-18\ ms$ in order to limit the jitter. This actually means having more capacity than needed when the system manages only one call.

For these reasons, we advice to consider the efficiency as a relative value to compare the multiple allocation methods. We observe that there are two groups. *RBDC + Min* is clearly not very efficient as the minimum allocation is not used in multiple cases. Given the nature of PTT conversations, the system does not receive calls in some phases of the simulations. The situation improves as more groups are simulated but remains far from the other methods. In the other group, *CRA PTT Rate* is the most efficient

RESTRICTED DOCUMENT

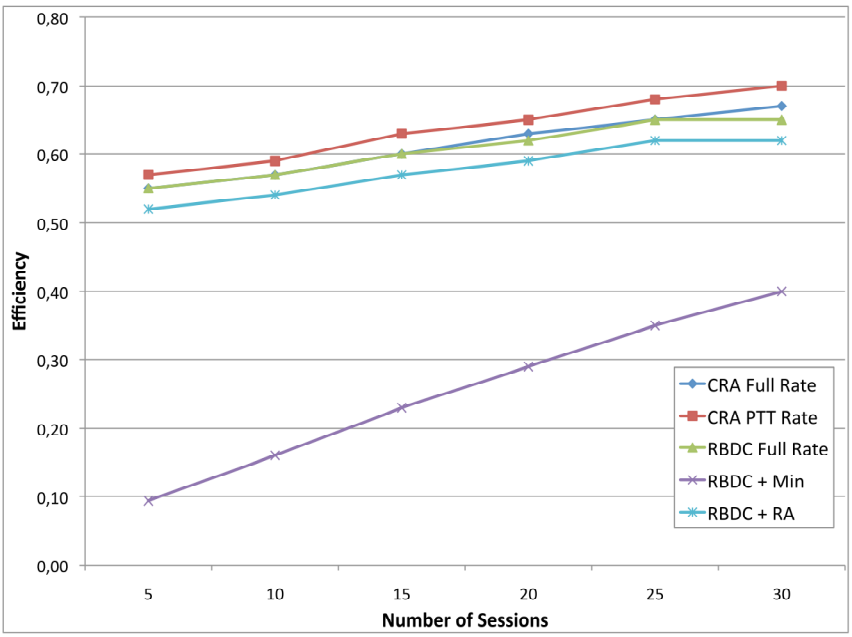


Figure 5.14: Efficiency

because it is the one that requests less resources, so less is wasted because of the voice activity factor, the ON-OFF pattern. However, this method needs volume allocations even more than the others to reduce the delay. Finally, *RBDC + RA* just uses the random access when it has packets to send, so efficiency improves compared to *RBDC + Min*.

In figure 5.15 we observe the resulting average queuing delay for the same scenario. First we observe that the *CRA PTT Rate*, based on a predicted rate is clearly worse than the others in terms of delay. Given the burstiness of the conversations, this method has to recourse to the volume demand, *VBDC*, in order to catch up. Even though it is allocated the corresponding average rate throughout a conversation, the variability of the arrivals makes it always lie behind the rest.

We observe that *CRA Full Rate* and *RBDC Full Rate* cases present small differences between them, the latter being slightly better. The queuing delay for these proposals improves as the number of sessions increases. This is caused by the increased probability that a call ends and just another arrives and can use its resources right away.

Finally, *RBDC + Min* and *RBDC + RA* are undeniably the best. The extra capacity given by the minimum allocation or the random access allows to manage the case when multiple calls arrive and we have allocation for some of them. Random access is better because it can actually use more time slots. The extra capacity is measured in slots instead of actual rate, which makes it better than the case with minimum allocation.

To conclude, figure 5.16 presents the results for the jitter caused by the queuing delay, without considering the other sources of jitter. We take the standard deviation of the jitter as measure. The histogram of the jitter recalls a gaussian distribution centered at zero and the standard deviation is a good value to

RESTRICTED DOCUMENT

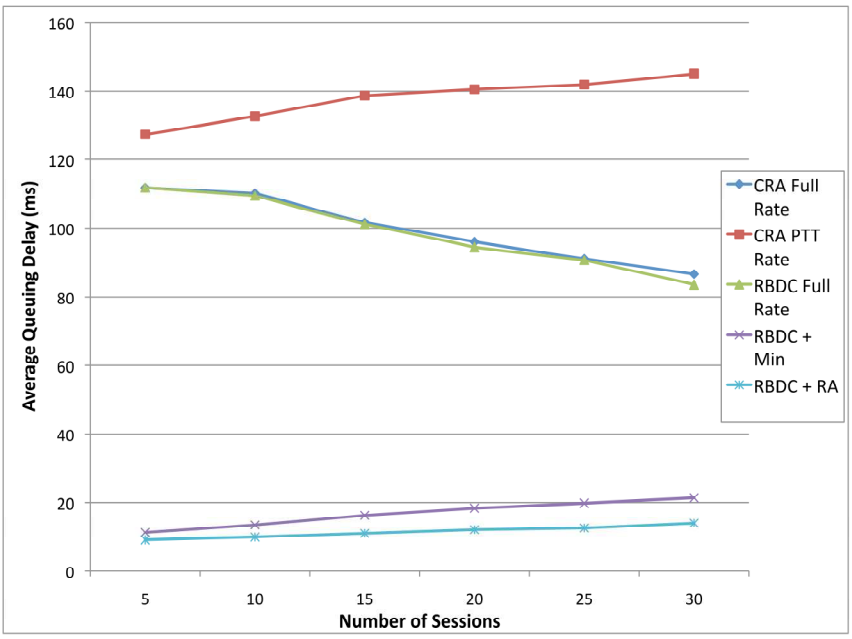


Figure 5.15: Queuing Delay

observe the necessary jitter buffer at the end user.

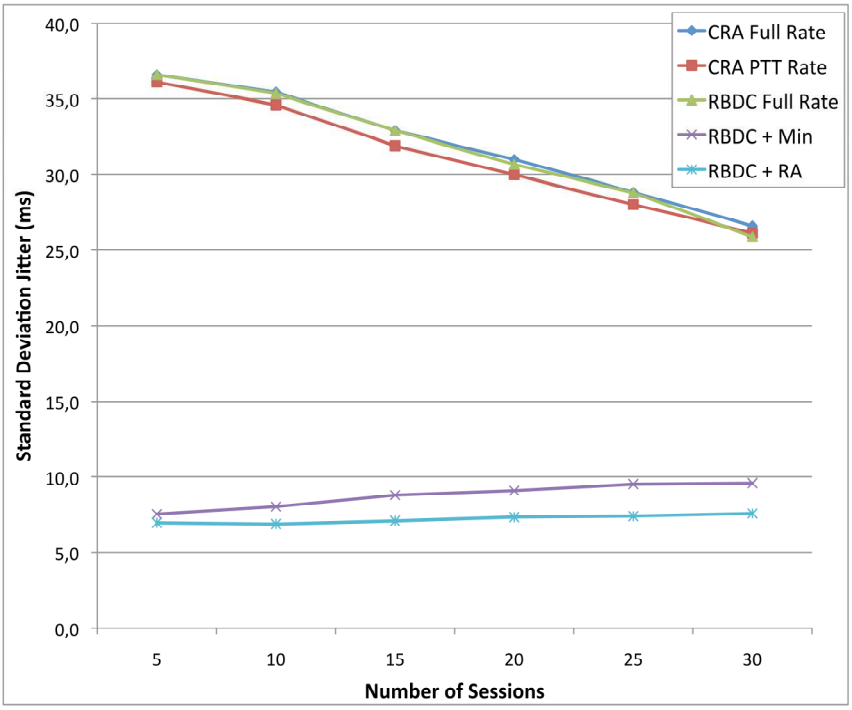


Figure 5.16: Standard Deviation of the jitter

Again, we have undoubtedly two groups. In the cases without an extra capacity the delay varies largely and it is necessary to have a large jitter buffer. The situation improves as the number of sessions

RESTRICTED DOCUMENT

increases due to the likelihood of unchaining multiple calls consecutively. The other methods, using a minimum extra capacity or the random access, perform very well and keep jitter standard deviation below the 10 *ms* mark.

After analyzing these results, we conclude that the best tradeoff of efficiency, delay and jitter is to use the *RBDC Full Rate + RA* option. Predictive rate methods outcomes are not very positive and we confirm our hypothesis. The preferable method is a one that relies on rate capacity demand for each call and benefits from an extra capacity via random access. The case with a minimum allocated bandwidth yielded good results as well in terms of delay and jitter. However, as the actual arrival rate of new conversation is low, even with a large number of groups, the efficiency is very low.

The simulated model considered that all calls were originated from the satellite cell, i.e. used the satellite return link. In a real scenario, we would have calls coming from either side of the satellite and methods employing a *CRA* allocation would suffer in efficiency. Mechanisms based on *RBDC* work call by call and can operate well in all situations.

RESTRICTED DOCUMENT

Chapter 6

Optimization of the user data transmission

6.1 Introduction

The establishment of the packet-based networks was a major breakthrough for voice communication services. The adoption of packetized voice allowed multiple users to share a common channel and, thus, increase the number of served clients without diminishing their quality of experience. However, sending voice in form of packets brings new issues as well.

Every voice frame that is packetized is preceded by a header. Voice packets are usually small compared to other types of information but the header itself remains similar. Therefore, the portion of the packet that contains the header information can be larger that the one that comprises the voice. This overhead affects the final throughput of the link, reducing the number of flows or users that can be served simultaneously.

Typically, voice is sent with a RTP header (12 bytes), an UDP header (8 bytes) and IP header (20 bytes for IPv4 or 60 bytes for IPv6). The payload size, depending on the coding protocol, can be as low as 12 bytes (case for AMR with 4,75 kbps of bit rate). Classic values are between 20 and 160 bytes, depending on the bit rate and the packetization period. Consequently, the final overhead introduced by the series of headers can be more than twice the actual payload size. Finally, it is important to recall that VoIP applications send typically 30-50 packets per second, aggravating the overhead effect.

Two main techniques are used to reduce the impact of this overhead: frame multiplexing and header compression. Multiplexing gathers multiple voice frames, i.e. payloads, in a single packet, reducing the weight of the header in the total packet size. On the other hand, header compression exploits the redundancy of some of the header fields and reduces the total information sent by consecutive packets.

Our approach is to make use of both techniques in order to reduce the number of bits needed to transmit one VoIP session, trying to increase the number of users while maintaining a good quality of service.

We see PTT voice as a form of VoIP, so we seek to apply the same techniques, with a few differences, to both services because they are likely to be used at a same time in a deployed LTE cell connected with a satellite link.

6.2 VoIP Multiplexing

VoIP carries a significant overhead, wasting useful bandwidth resources with every packet transmission. One possible solution is to increase the number of voice frames that a single packet transports. In this way, a single header can provide the necessary information to process multiple frames, thus increasing the throughput efficiency.

A multiplexer (MUX) intercepts multiple packets containing one voice frame and generates a single packet. On the opposite side, a demultiplexer (DEMUX), does the inverse job and retrieves the original packets.

Multiplexing can be performed in two ways. Intra-flow aggregation refers to the case when multiple frames from the same call, or flow, are carried in a packet. In contrast, inter-flow aggregation combines frames from different calls into the same packet. In the latter, usually a mini header is added in order to identify the flow to which the voice frame belongs to. Intra-flow aggregation can be performed end-to-end, from user to user, increasing the overall efficiency. However, it introduces an important delay, as the multiplexer has to wait for the voice frames to be created. While it necessary that more calls travel on the same path, inter-flow multiplexing reduces the introduced delay as the multiple frames arrive closer in time. This approach is more efficiently performed hop-by-hop, as the multiplexing entities may not be as close to the users like in the first case. Figures 6.1 and 6.2 provide an overview of these two options.

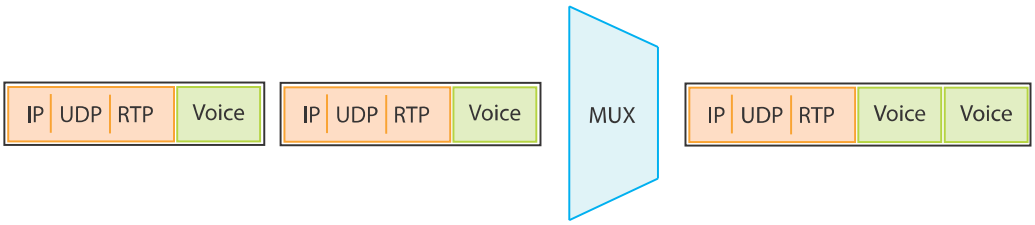


Figure 6.1: Intra-Flow Aggregation

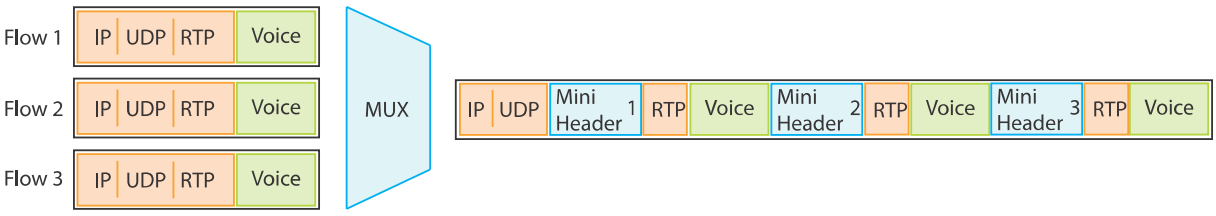


Figure 6.2: Inter-Flow Aggregation

Additionally, it is possible to perform the frame aggregation in multiple layers. The higher the multiplexing layer, the more efficiency is achieved. Many voice codecs are able to perform intra-flow aggregation directly, allowing aggregation at the RTP layer, as it can be seen in figure 6.1. In the inter-flow case, the aggregation is generally performed in the IP or UDP layers. Figure 6.2 shows an example for multiplexing at the UDP layer.

RESTRICTED DOCUMENT

In order to decide how many frames to aggregate there exists two limiting parameters that the multiplexing algorithm needs to take into account. The first one is the maximum number of frames that can be combined in a single packet, usually named as Nmax. Its value depends strongly on the maximum size allowed in the lower layers, the maximum transmission unit or MTU, but also on the size of the frame themselves and the multiplexing layer, which defines the headers that are needed. The second one refers to the maximum waiting time Tmax, or retention time, i.e. the longest period of time that a frame can be delayed in the multiplexer buffer. In the inter-flow approach, this is limited by the time between two frames of the same call, the transmission time interval or TTI, in order to avoid having two frames of the same flow in a multiplexed packet.

Then, an aggregated frame is transmitted when the number of multiplexed frames reaches Nmax or the time since the arrival of the first voice frames attains Tmax. The value of such parameters depends on the actual implementation and adaptive algorithms can be applied.

One should bear in mind that higher aggregation ratio and thus lower overhead does not necessarily mean better throughput. Larger packets are more likely to be corrupted by bit errors or interferences.

Multiplexing has been an extensive field of study in VoIP optimization. The approaches defer depending on the lower layers used, the signaling protocol or the service provided. [99] presents a multiplexing scheme for H.323 VoIP services that also focuses on header compression. [100] proposes a technique for multiplexing on the UDP layer. Their innovating mechanism reduces the size of the packets by transmitting only the difference between consecutive packets. [101] combines multiplexing and compression of the voice frames to increase overall capacity.

Other works study the performance on wireless LANs. For example, [102] targets the downlink of a WLAN and combines multiple streams into a single packet for multicast. Then, all stations should be capable to process those multicast packets. As they highlight, encryption of the voice frames should be performed for security reasons. Some studies focus on ad-hoc wireless networks, where every station may act as a (de)multiplexor and (de)aggregates frames coming from or going to different paths. Finally, [103] proposes and compares a series of multiplexing algorithms, adapting Nmax and Tmax, for multiplexing in the LTE backhaul. Therefore, their target link is the connection between the eNBs and the P-GW.

6.3 Header Compression

The headers of the network, transport and session protocols incorporate a number of information fields that allow the entities on the same layer to understand, for example, the information within the payload, who is the intended recipient or who is sending the packet. However, when analyzing a single session, most of these fields are either static or rarely changing. Hence, there is an opportunity to compress the information contained in those headers by suppressing some of the fields or linking their value to the evolution of others, like the RTP sequence number. Then, on the other end of a link, a decompressor is able to interpret the information received and reproduce the packet as it was originally. Not all the

RESTRICTED DOCUMENT

protocols aim at compressing the headers of multiple layers. Some consider only the IP layer while others focus on the RTP session headers.

Several protocols for header compression have been proposed. Some of the most common algorithms are compressed RTP [104], enhanced compressed RTP [105], IP Header Compression [106] and Robust Header Compression (RoHC) [107]. Next we review RoHC more extensively because our protocol is based on the techniques introduced by it. While RoHC requires considerably more processing and memory resources, it performs better over links with high bit error rate such as wireless links.

6.3.1 Robust Header Compression (RoHC)

Header fields have a significant redundancy between them, either among fields within the same header or between consecutive packets of the same stream. Therefore, the static information can be sent initially and by exploiting the dependencies between some fields, the header size can be reduced. All relevant information concerning a packet stream is saved under the same context and is used to process, compress or decompress, the subsequent headers. Each context is assigned an ID number that allows the decompressor to identify the stream to which the received header belongs to. This context is updated depending on the values and parameters received and it is important that both compressor and decompressor maintain the common context well updated in order to correctly process the headers. The loss of one or multiple packets may lead to inconsistencies between the context of the different entities. The techniques adopted by RoHC aim at reducing the effect of the losses and improve the robustness of the compression algorithm.

RoHC provides different profiles depending on the headers that are compressed. For example, the RTP profile compresses IP/UDP/RTP, the UDP one IP/UDP, and the IP one only compresses the IP headers. Other profiles also exist to be able to send whatever packet, through an uncompressed profile. Other profiles have been introduced in the last decade.

The header fields are classified and treated differently by the compressing algorithm:

- *Static:* Fields that are expected to remain constant on all packets and are only needed to be communicated once.
- *Static-Def:* These are static fields that define a packet stream.
- *Static-Known:* Static fields that have values that are well known for the compressor and decompressor and it is not necessary to transmit them.
- *Inferred:* These values can be inferred from other values, such as the size of the frame, and therefore can be left out by the compression mechanism.
- *Changing:* Finally, the fields that change their value over subsequent headers.

When considering the fields that are actually changing, they can be further classified in different subclasses:

- *Static:* Fields that change regularly and follow a known pattern.
- *Semistatic:* Fields that are static most of time but that occasionally change and then come back to the usual value.
- *Rarely-Changing:* They change occasionally and when they do, they tend to keep the new adopted value.
- *Alternating:* They alternate between a few number of values.
- *Irregular:* They seem to change randomly or under a non identified pattern.

Given these classifications, some of the fields are not transmitted at all because they are inferred from the profile information or the packet length; the ones that define the stream are transmitted only at the beginning; the ones that change occasionally are transmitted initially but prepared to update; the rarely changing should be updated or even sent as-is frequently; the identified with a random changing pattern should be sent as is; the RTP sequence number, and the IPv4 id if set sequentially, should be encoded using a fixed delta and they should be robust towards packet losses; finally, the RTP timestamp is usually changing by a fixed delta but it can be necessary to update it.

Most of fields can be set at the start of the flow and only be updated occasionally. Only some fields require further attention. The values of the RTP Timestamp and the IPv4 can usually be predicted from the RTP Sequence Number. The RTP marker is usually constant, but may be need to be communicated when it changes, but it is only one bit. Finally, if used, the UDP checksum cannot be predicted and it should be sent completely.

The main principle is to establish the relationship between the RTP sequence number and the other changing fields. Then, the RTP should be transmitted reliably and if the function that allows retrieving some other field from the sequence number changes, additional information is sent to update the context.

RoHC Compression/Decompression states

For each context, two state machines are maintained, one at the compressor and another one at the decompressor. The transition between states does not need to be synchronized between the two machines. We first look at the state machine of the compressor at figure 6.3.

Figure 6.3: Compressor State Machine

In the initialization and refresh state, the compressor sends a header with the complete context information. The headers within the first order state include all the relevant information about changing fields

RESTRICTED DOCUMENT

and serve to communicate change of patterns or irregularities. Finally, second order state reaches the optimal compression where most of the fields are static or can be predicted from the sequence number.

The decision to transit between states can be consequence of changes in the header fields, receiving either positive or negative feedback from the decompressor, or by following periodic timeouts, when the unidirectional mode is used. Next we observe the machine of the decompressor in figure 6.4.

Figure 6.4: Decompressor State Machine

In the no context state, the decompressor is not able to decompress any packet. In static context, the dynamic part has been lost or is missing and requires at least a FO-state header to evolve to the full context state where all headers can be decompressed.

RoHC Operation modes

Depending on the characteristics of the link and the processing capabilities, three different operation modes have been designed:

- *Unidirectional (U-Mode):* This mode is used when no feedback is possible or when it is preferred that packets are sent in one direction only. The transitions between compressor states are triggered by periodic timers and field pattern changes. This mode is less efficient and robust against propagation losses. IR packets are then periodically sent to maintain the context.
- *Bidirectional Optimistic (O-Mode):* The return channel is used to send negative feedback leading to error recovery requests and, optionally, to acknowledge significant context updates. Periodic refreshes are generally not used and only sent if necessary. Compression efficiency and robustness are improved. Nevertheless, the possibility of context invalidation may be higher than in R-Mode because of the low usage of the feedback channel.
- *Bidirectional Reliable (R-Mode):* The feedback channel is used extensively upon all context updates, including updates of the sequence number. This mode maximizes the robustness against errors, minimizing the probability of context invalidation.

RoHC encoding methods

Early compression schemes introduced the delta-encoding method to compress some of the information fields in consecutive packets. The compressor sends an initialization header with the full information regarding the different fields. As we have seen before, most of the changing fields follow a static pattern. Therefore, when using the delta-enconding mechanism, the compressor would only send the difference

RESTRICTED DOCUMENT

between the fields in the consecutive headers. It is an extremely simple method and it allows compressing some fields considerably, given that they usually change only by one. Yet, despite its simplicity, it is unable to properly handle packet losses before and after compression. It does not introduce a robustness mechanism that prevents loss propagation after one packet is lost between the compressor and the decompressor.

One of the main principles of RoHC is to attack this issue. RoHC was designed to be used with wireless links, where packet losses are usually common. RoHC introduces a more complex and robust algorithm called Window-based Least Significant Bit (W-LSB).

Least Significant Bits encoding Simple LSB encoding for the transmission of year values can be observed in figure 6.5. Imagine that you want to transmit all the years corresponding to the 20th century. As the two first digits, 19, are common for all years, the compressor only sends the ones corresponding to the least significant. The decompressor will later be able to retrieve all the years even if any of them is missing. Both entities will have stored in their context the beginning of the century, 1900. Observe that even if a packet is missing, the next one can be encoded with the same number of bits. Nevertheless, packet reordering and packet losses beyond the window size, 10 in the year example, cannot be handled and the main challenge is to properly advance the window reference for both compressor and decompressor.

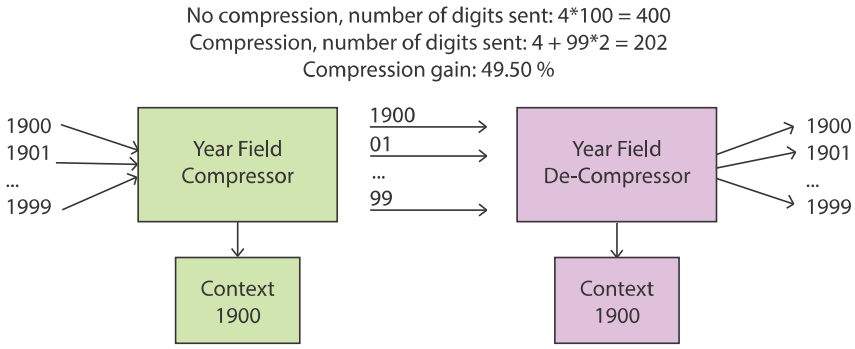


Figure 6.5: Year Least Significant Bit encoding

LSB encoding helps compressing fields with small changes transmitting only the K least significant bits. Given a value of reference, v_ref, we can define an interpretation interval. The values within this interval, observed in figure 6.6, can be encoded considering the value of reference and transmitting up to K bits.

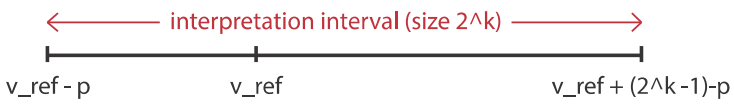


Figure 6.6: Interpretation Interval for the Least Significant Bit encoding

We can define an interpretation interval function englobing the interval of numbers we can encode:

RESTRICTED DOCUMENT

$$f(v\_ref, K) \quad = \quad [v\_ref - p, v\_ref + (2^K - 1) - p] \tag{6.1}$$

p is defined as the interpretation interval shift. Depending on the change pattern of the encoded field, p could take different values:

- Fields that are always expected to increase: $p = -1$ and $interval = [v\_ref + 1, v\_ref + 2^K]$.
- Fields that stay the same or increase: $p = 0$ and $interval = [v\_ref, v\_ref + 2^K - 1]$.
- Fields expected to deviate only slightly from a constant reference value: $p = 2^{K-1} - 1$ and $interval = [v\_ref - 2^{K-1} + 1, v\_ref + 2^{K-1} - 1]$.
- Fields expected to undergo small negative changes and larger positive: $p = 2^{K-2} - 1$ and $interval = [v\_ref - 2^{K-2} + 1, v\_ref + 3 * 2^{K-2}]$.

The compression and decompression procedure is described next:

- The compressor (decompressor) always uses v_ref_c (v_ref_d), the last value that has been compressed (decompressed), as reference.
- When compressing a value v, the compressor uses the smallest value of K such that v lies within the interval defined by the interpretation interval function.
- When receiving m LSBs, the decompressor uses its own interval function defined by $f(v\_ref, m)$ and picks as decompressed value the one that matches the received LSBs in that interval.

In order to make sure the reference has been well received, they are protected with a cyclic redundant code (CRC). The compressor uses the last value protected as reference and the decompressor uses the last decompressed value that was CRC protected.

Window-based Least Significant Bits encoding The W-LSB algorithm works a bit differently. Instead of relying on a static reference value, the reference changes according to a sliding window, reducing even more the number of bits needed to transmit the whole century, as we see in figure 6.7.

Sometimes, it is difficult to determine by the compressor which value is currently used as reference in the decompressor. Hence, the innovation introduced by RoHC is to calculate the number of bits K that assures that, independently of the current value of reference v_ref_d, the value to be compressed lies within the decompression interval. The compressor maintains a sliding window of possible candidates for v_ref_d. The process can be summarized as follows:

- Upon transmission of a value v protected with a CRC, the compressor adds v to the sliding window of candidates.

RESTRICTED DOCUMENT

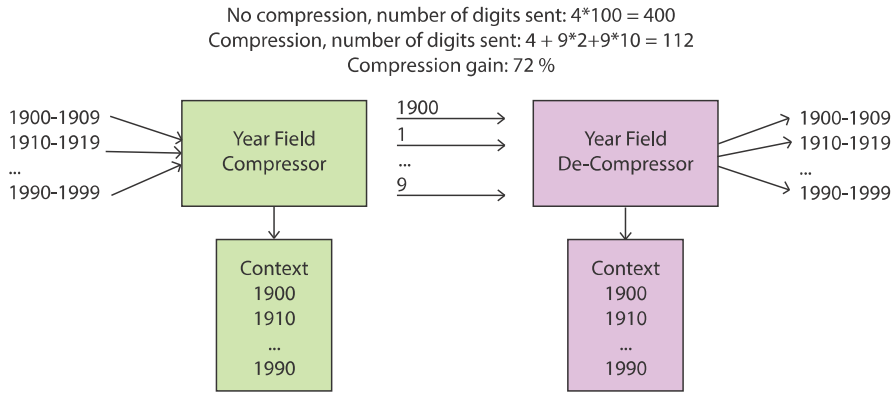


Figure 6.7: Year Window-based Least Significant Bit encoding

- For each value v to be compressed, the compressor chooses $K = max(g(v\_min, v), g(v\_max, v))$; where g is the inverse function of f.
- When the compressor gains enough confidence that a certain value v and the previous ones are not used as reference anymore, the sliding window progresses and removes these values.

Scaled RTP Timestamp encoding One of the options RoHC proposes to compress the RTP Timestamp is the scaled encoding. In voice and video applications, the timestamp between consecutive RTP packets is expected to increase some constant number that RoHC identifies as TS_stride. This number is usually a function of the number of milliseconds of audio or video included in a frame and the sampling rate. Then, to encode the timestamp (TS), the compressor downscales it by a factor of TS_stride, saving $floor(log_2(TS\_stride))$ bits in each transmission. The value can be retrieved with the following formula:

$$TS = TS\_scaled * TS\_stride + TS\_offset \tag{6.2}$$

TS_stride is explicitly communicated to the decompressor and in its turn, TS_offset is implicitly communicated as the TS progresses. TS_scaled will then follow the pattern of the RTP sequence number that is encoded with the W-LSB method. The process is resumed next:

- *Initialization:* The compressor sends TS_stride and the complete value of one or several TS fields to initialize TS_offset.
- *Compression:* The compressor will calculate the TS_scaled using $TS\_scaled = \frac{TS - TS\_offset}{TS\_stride}$. If it follows the same pattern as the sequence number, it does not need to be specified explicitly.
- *Decompression:* The decompressor derives the original TS_scaled from the RTP sequence number and computes the original value: $TS = TS\_scaled * TS\_stride + TS\_offset$.
- *Offset at wraparound:* When the unscaled TS reaches the maximum value, given that is defined by a 32-bit field, the current value of TS_offset will be invalided and a new value needs to be

RESTRICTED DOCUMENT

obtained: $TS\_offset = (WrappedaroundunscaledTS) mod (TS\_stride)$.

- *Interpretation interval at wraparound:* As observed in figure 6.8, the maximum scaled TS, TSS_max, may not have the form of $2^K - 1$. When TSS_max is part of the interpretation interval, a number of unused values are inserted such that their LSBs follow the usual path. The number of LSBs used increases to disambiguate between TSS_max and 0 when the LSBs of TSS_max are set to zero.

Figure 6.8: Interpretation interval at wraparound

Encoding of self-describing variable-length values The values of context IDs and TS_stride may be described by a variable number of bits (bytes) depending on its value. In order to indicate to the decompressor the number of bits that it has to read to retrieve the given field, the following encoding method is used:

- To communicate numbers up to 127: 1 byte is transmitted, whose first bit is 0. 7 bits can be used to indicate the actual value.
- To communicate numbers up to 16383: 2 bytes are transmitted, whose first bits are 10. 14 bits can be used to indicate the actual value.
- To communicate numbers up to 2097151: 3 bytes are transmitted, whose first bits are 110. 21 bits can be used to indicate the actual value.
- To communicate numbers up to 536870911: 4 bytes are transmitted, whose first bits are 111. 29 bits can be used to indicate the actual value.

RoHC over channels that can reorder packets

The channels where RoHC is used could suffer from packet losses as well as reordering issues. Reordering makes packets arrive in a different order they were conceived, the order indicated by the sequence number. Regarding the encoding of the RTP sequence number with the W-LSB method, one could rethink about the interpretation interval and make the compression robust against reordering. Observe figure 6.9. On one hand, the ability to decompress a sequentially late packet is limited by the offset p of the interpretation interval. On the other hand, the decompressor will be able to decompress a sequentially early as long as it is inside the part of the interval corresponding to the protection against losses, $[v\_ref, v\_ref + (2^K - 1) - p]$. At the moment of arrival of this packet, the packets in between that are delayed cannot be differentiated from lost packets.

RESTRICTED DOCUMENT

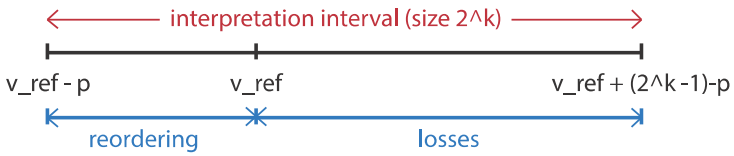


Figure 6.9: Robustness against reordering and packet losses

The problem appears when the reordering happens in a moment that involves change packets, where some fields have changed either their value or their changing pattern. A sequentially late packet with respect to a change packet may not be decompressed because the context information for successful decompression may not be available anymore.

The successful decompression of a sequentially late packet may cause the decompressor to update the context in an unexpected manner and lead to a loss of syntonization. Generally, as the outcome of the decompression of updating packets can be verified, the decompressor can reliably detect decompression failures and discard the packet.

Increasing the tolerance against reordering Compressor implementation could adjust its optimistic approach and think about changing its strategy to increase the protection against both reordering and losses. For example, the number of repetitions of context update messages could be increased. Additionally, the compressor could consistently use packet formats that use a larger number of LSBs to communicate the sequence number update allowing to use a larger negative offset. Hence, the capacity to decompress sequentially late packets increases considerably.

It is a tradeoff between a more aggressive implementation with better compression efficiency and another one, better prepared to handle the possible reordering and losses within the channel.

It could be possible to attempt to decompress sequentially late packets by going backwards in the interpretation interval and re-using reference values that were supposed to be obsolete. After successful decompression, the packet could be handled to the upper layers without updating the context.

Finally, new RoHC profiles could define new values for the p offset in order to achieve a robustness against reordering similar to the effect of selecting packet types with larger number of LSBs. The tradeoff is at the expense of losing robustness against packet losses. For example, table 6.1 shows the typical values reflecting this tradeoff and in table 6.2 we observe an alternative choice that would increase the protection against reordering.

6.4 Towards a robust and compressed multiplexing scheme

Typical multiplexing mechanisms that create packets such as the one shown in figure 6.2 feature fixed-size mini-headers in addition to full size RTP headers. Mini-headers include a context identification number that allows the receiver to properly classify the frame and transform it into a regular packet. However, it is expected that when a new transmission begins, the initialization of the context should be

RESTRICTED DOCUMENT

k (bits SN)	Offset p (reordering)	$(2^k - 1) - p$ (losses)
4	1	14
5	0	31
6	1	62
7	3	124
8	7	248
9	15	496

Table 6.1: Classic tradeoff for robustness against reordering and packet losses

k (bits SN)	Offset p (reordering)	$(2^k - 1) - p$ (losses)
4	5	10
5	10	21
6	21	42
7	42	85
8	85	170
9	170	341

Table 6.2: Options to increase robustness against packet reordering

RESTRICTED DOCUMENT

handled in separate fashion. In the rarely event that some fields in the IP-UDP headers change, they shall be updated in a similar manner. If we consider that the context ID is at least 1 byte, the mini header would be somehow larger and together with the RTP header, the multiplexer needs to send around 10 bytes for each frame.

Instead of using mini-headers and full RTP headers, we propose to use compressed headers, just like in RoHC. Most frames can be sent together with a compressed header of only 2 bytes. Additionally, initialization and update headers can be sent into the MUX packet like any other frame. We consider that some of the fields in the IP are not useful and therefore those initialization headers can be reduced in size.

The only issue that appeared at the beginning was that RoHC needs another layer to provide the frame size. However, considering a voice application, the size of the payload rarely changes and therefore, we can infer it from the IP header part of the initialization and update header.

The decompression operation is then the same as in RoHC. The decompressor analyses frame after frame to retrieve the original packet. It is possible that it is not able to decompress a frame, then it shall discard it and proceed to the next one. As the decompressor knows the typical sizes of both headers and payloads, it can attempt to decompress the following frame. We expect the number of possible payload sizes, i.e. voice application rates, to be limited. Hence, one error does not mean discarding the rest of the packet.

Compression follows again the RoHC operation, changing its compression state just like in the uni-directional mode. It could be possible to compress also the outer headers, the ones of the MUX packet, in the satellite terminal if the gateway-terminal pair feature a compressor-decompressor mechanism. The reason we decide to do a MUX packet with compressed mini-headers was to avoid the hop-by-hop compression-decompression. Hence, the compression is done once in the first super node and it continues as this until it arrives to the destination super node.

The following subsection details the used headers, borrowed mainly from the second version of RoHC [108].

6.4.1 Header types

We consider mainly three different headers: initalization/update, burst update and full compressed. The first one includes most of the fields of the IP+UDP+RTP chain, uncompressed. The second is useful at the beginning of a new talkspurt, after a silence period. It is necessary to update the timestamp parameters to take this silence into account and allow the decompressor to pick up with the operation. Finally, there is the full compressed packet, including only a compressed sequence number and a CRC (cyclic redundant code) for verification.

The headers used are the same featured in RoHC version 2, except the initialization one, that does not send some of the fields. Considering the context ID, we limited the number of possible connections between a pair of super nodes to 256. In that way, it can be indicated with only one byte. To extent this, one could follow the RoHC standard and use self-encoding values and put the first byte first and then the

RESTRICTED DOCUMENT

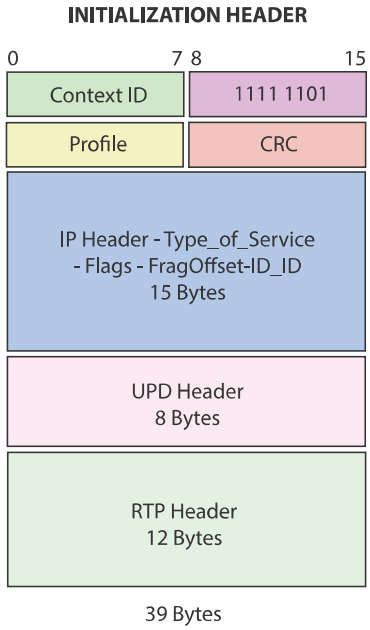


Figure 6.10: Initialization Header

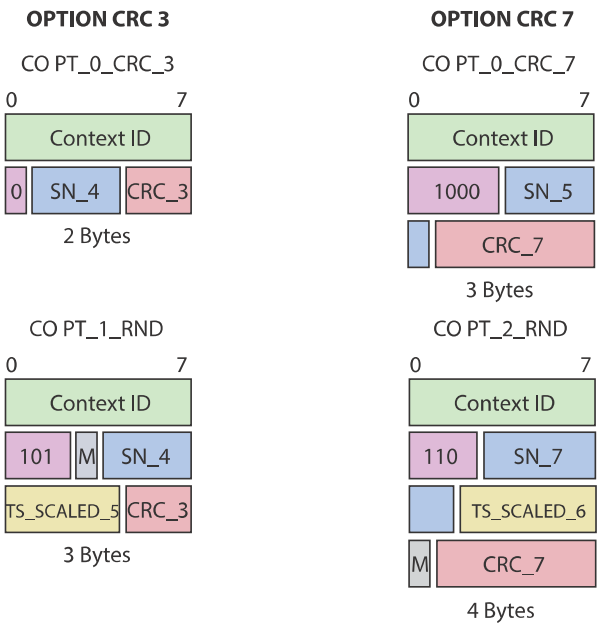


Figure 6.11: Compressed Headers options

rest after the second byte of the header, the one that incorporates the fixed-value of the type of header.

Figure 6.10 shows the initialization header. It includes the context ID, the header type indication (1111 1101), the profile and the CRC for this part. We could use the same profile number as the RTP profile in RoHC, only if we are sure that both super nodes are well informed that the initialization header has changed. Then, the header includes the original IP+UDP+RTP chain.

We consider that some fields are not really useful in this context. The type of service is the same as the one included in the outer IP header. As we forbid to fragment the frames into the MUX, the associated flags and fragmentation offset options are not useful either. Finally, IP ID does not provide any relevant information. However, in order to calculate the IP checksum, we think that the best is to use the same values always for the IP ID, known both by the compressor and the decompressor.

In figure 6.11 we observe the considered compressed headers. We consider two possible options, with a CRC of 3 bits and with a CRC of 7 bits. The latter headers are one byte larger and allow to send larger sequence number and, therefore, provide better robustness against reordering or losses. In our simulation we used the CRC-3 headers.

The upper part of the figure shows the full compressed headers (packet type 0, pt_0) that include only the context id, the header type identification, the compressed sequence number and the CRC. If we recover the previous discussion about the robustness against reordering and errors, we consider that the option using SNs of 4 bits, takes $p = 1$ and $(2^k - 1) - p = 14$. In the case with SNs of 5 bits, we consider $p = 3$ and $(2^k - 1) - p = 28$.

The lower part of the figure shows the update talkspurt headers (pt_1_rnd and pt_2_rnd). These headers include the M field and the compressed scaled Timestamp in order to help the decompressor

RESTRICTED DOCUMENT

with the interpretation of the timestamp.

We consider that these three header types are sufficient for a voice application. In case of a change in the codec, the best is to restart the compression operation and send some initialization packets.

6.5 Simulation and Performance Evaluation

6.5.1 Scenario

We developed our solution in the event-driven network simulator ns-3 [83]. Our purpose was to perform a capacity estimation for a given scenario. We compared the baseline case, without neither compression nor multiplexing, to our proposed solution.

Figure 6.12 gives an overview of the tested scenario. First, a traffic generator was implemented. Each source follows an on-off pattern with exponential distribution. We decided to use a 16 kb/s codec that sends a voice frame every 20 ms in ON periods. Then, traffic goes through a link that simulates an LTE cell. We use a jitter model derived from the measurements performed in [109] for voice over LTE. After, the satellite terminal features the compressor and multiplexer. The satellite channel was specially designed and emulates a DVB-S2 link in clear sky with different modulation and coding configurations. The demultiplexer and decompressor are located at the reception of the satellite system. Next, a link simulates the core network. Finally, a sink node receives all the frames and emulates an equivalent number of players with a jitter buffer.

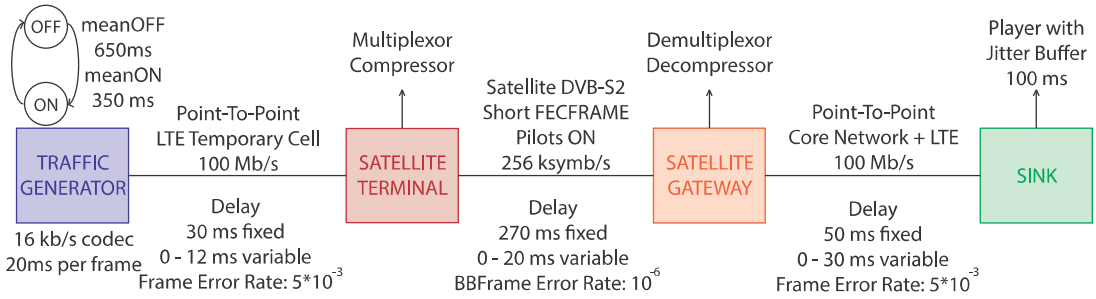


Figure 6.12: Simulated Scenario

In order to derive the maximum number of concurrent sessions that the system can manage, a set of Quality of Service (QoS) requirements was set. First, the average end-to-end delay was limited to 450 ms with maximum values up to 500 ms. Secondly, the maximum admitted jitter was 100 ms and its standard deviation, measuring the dispersion of values, was set below 15 ms. Finally, the considered maximum end-to-end frame loss rate limit was positioned at 3 %. Under this limit, the presence of large burst of errors that strongly impact the user quality of experience is uncommon.

RESTRICTED DOCUMENT

6.5.2 Results with Compressed MUX scheme

This subsection presents how our solution performs against the baseline design before applying any optimization. Figure 6.13 shows the number of supported sessions for both cases. We can see that the proposed scheme almost doubles the capacity when we use a ModCod with high spectral efficiency. Even in the most constrained case, the improvement is more than 60%.

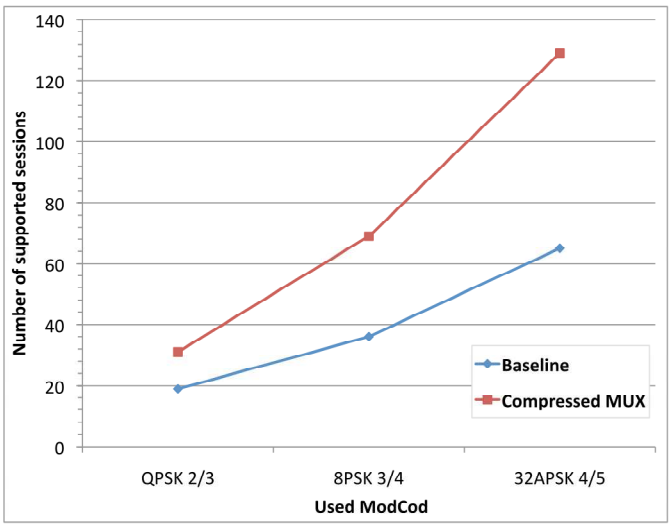


Figure 6.13: Baseline vs Compressed MUX

Although these results are very positive, using a header compression mechanism has some drawbacks. While the baseline case was almost free of long error bursts, with more than three packets lost in a row we observe that some sessions experience bursts of errors. With the proposed QoS requirements, in the baseline case, almost all users would be very satisfied. However, in the compressed MUX case, some users would be less satisfied. Our simulations last for 200 seconds but a typical PTT message would be in the range of 10 seconds. So the probability of having a slightly dissatisfying message would be inferior.

These error bursts are caused by the succession of multiple decompression errors. When the decompressor is not able to correctly process a header, it drops all the frames of the same session until a full header is received. For this reason, we restrained from increasing the compression ratio excessively. We use a 2/10 ratio, i.e. 2 uncompressed headers followed by 10 compressed ones in a given session.

It is interesting to observe the effect of using different compression ratios. Figure 6.14 shows these for the case of 8PSK with the maximum allowed number of users. We notice that increasing the ratio reduces the overall frame error rate. Nevertheless, we see how the decompression error increases until a point that makes the total error rate to go up again. This will increase error bursts and their duration as well; therefore, we do not encourage using a very high compression ratio in similar scenarios.

RESTRICTED DOCUMENT

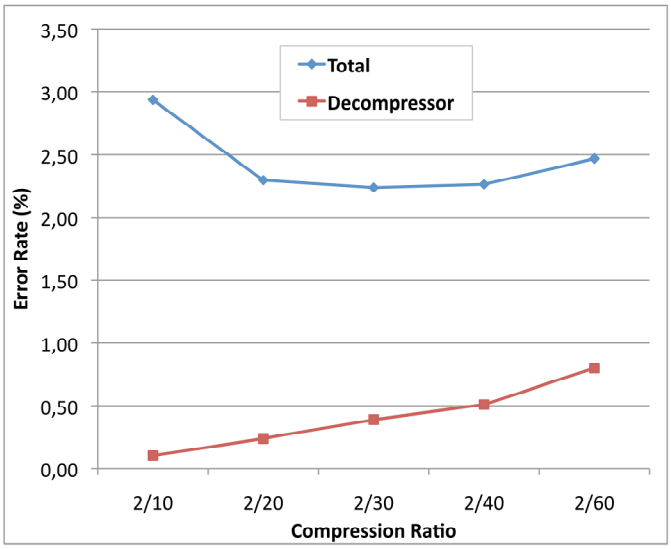


Figure 6.14: Effects of Compression Ratio

RESTRICTED DOCUMENT

Chapter 7

VoIP and PTT Service Differentiation

7.1 Introduction

The solution that we presented in the previous chapter considers the processing of VoIP calls and PTT messages together. Both are voice services that are considered premium in the scenario of temporarily deployed cellular networks. However, there is a small difference, PTT is unidirectional. As there is no necessity to synchronize with a flow that comes in the opposite direction, PTT works well even with a larger delay, as long as the message can be played smoothly.

Therefore, there is an opportunity to increase the number of concurrent sessions even further than using only header compression, by applying a delay service differentiation between these applications. The goal is to allow a higher multiplexing delay for the PTT application, while keeping the delay for VoIP calls as low as possible. A differentiation ratio can be then calculated by dividing the delay goal of both services. We expect that having a perfect differentiation will be difficult given the inherent wait between each transmission opportunity in the satellite system. Additionally, one should consider that several frames would depart at the same time within the same MUX packet.

Note that the dropping rate of both applications should be similar as they have the same requirements in respect to the robustness against the frame loss. In the remainder of this chapter we review three different possibilities to perform such differentiation. We first provide some background about diverse scheduling approaches, some of which also address the delay differentiation issue. The last section of this chapter is devoted to the comparison between the selected three scheduling mechanisms.

7.1.1 Background on Scheduling

A terminal or a router likely has multiple packets waiting to be transmitted to their corresponding recipients. Scheduling protocols decide which packet is sent in the next transmission opportunity. Depending on the physical layer available, packets can be sent one by one or multiple packets are sent together as part of a larger frame. It shall not be confused with buffer or queue management, which deals with the acceptance or rejection of new packets entering the node, or even removing some of the packets already

on the buffer.

We review the main goals and properties of such methods and then briefly discuss about some of the most common ones.

Main goals The main objective is to provide a set of service guarantees, such as bandwidth, delay, loss or jitter. Hence, by giving a flow enough transmission opportunities, the protocol ensures an upper bound on the maximum waiting time, limiting the dropping of the packets that are not within the bounds and maintaining the delay difference of consecutive packets sufficiently low. Additionally, another goal is to guarantee a fair share of the excess resources depending on the service requirements. Finally, an important issue is to minimize the processing load during the next-packet decision process and provide a good overview on whether a new flow can be accepted.

Properties Given the set of requirements of each service, the scheduling method shall guarantee the service to each flow independently of the behavior of the rest, this is known as flow isolation. Furthermore, as noticed above, the protocol needs to take advantage of the excess capacity and distribute it fairly among all flows. To conclude, implementation complexity and efficiency, measured by the number of flows that can be admitted given a set of requirements and available resources, allow comparing multiple different methods.

Typical Scheduling Protocols The first method is the rudimentary First-In First-Out (FIFO) where packets are simply served in the order of arrival. Unfortunately, there is no flow isolation nor any guarantees. Taking this approach, the concept of priority queues was introduced: multiple FIFO queues, one per type of service, and the scheduler selects the higher priority queues as long as they have packets to transmit. A priority level is guaranteed but the lowest class queues can be starved.

Probably the other most common family of schedulers follow the round-robin principles. Round-robin picks one packet from each queue and after all queues have been visited, the process restarts. In order to introduce a grade of differentiation between queues, weighted round robin assigns a weight or a share of the capacity to each queue. The transmitter visits each queue and transmits a number of packets equivalent to the percentage of the total packets sent in the last transmission opportunity or period. Nevertheless, the fairness of this scheduler is highly dependent on the number of connections or queues and the smaller weights.

However, given the variable size of packets, it was necessary to introduce an evolution, the Deficit Round Robin [110]. At each round, each queue is given a quantum that represents the total number of bits it should be sending at each round, depending on its weight. If the remaining number of bits do not account for the size of the first packet in the queue, this share of bits is left for the next round. Hence, in average, each queue will have the corresponding weight of the output capacity. There is not, however, guarantee for protecting any flow from malicious users. **Given its simplicity, we have adopted it and we focus on adapting the weights to achieve delay differentiation**.

RESTRICTED DOCUMENT

The other big family of schedulers inherit from the Fair Queuing principle, which in its turn is based on the concept of an idealized fluid flow model, the generalized processor sharing (GPS). Each flow is assigned a queue and a weight of the capacity. A scheduler can serve the smaller amount, a bit, of data at each visit and the frequency of visits is given by the weight, the amount served so far and the number of active flows. This model has to be adapted to a real scheduler where the transmission of packets cannot be interrupted, therefore it is necessary to compute the desired transmission time following the fluid model and selecting the packet with the smallest time. This scheduler receives the name of Weighted Fair Queuing (WFQ), which achieves fairness and bounded delay, but does not overcome the complexity problem. There exist many evolutions of this scheduler that improve the fairness and the delay bound but still require a high computational load. Probably, the most commonly known is the Worst Case Fair Weighted Fair Queuing (WF2Q). The transmitter selects the next packet from those that would started receiving service in the GPS model, the eligible packets.

These schedulers have the ability to provide a transmission rate to all individual flows independently of the behavior of the rest and to guarantee a fair access to the excess capacity. This allows to limit the local delay of each flow. However, this could be inefficient for low bandwidth flows. Finally, there is no possibility to control the jitter between consecutive packets.

Furthermore, another type of mechanisms is known as timestampted schedulers. Every packet has a corresponding timestamp that is the base of the decision process. The first of this kind is the Earliest Deadline First (EDF), where packets are assigned a deadline depending on their priority and then the queue is ordered by deadlines, i.e. the packet with the earliest deadline is chosen for transmission. Delay Earliest-Due-Date (Delay EDD) introduced an improvement by computing the deadline depending on the packet arrival time. Yet, it does not provide a sufficient bounding for the delay variation. Jitter Earliest-Due-Date (Jitter EDD) was finally proposed to overcome this issue and **it has been selected as one of our proposed solutions**.

Virtual Clock (VC) attempts to emulate a perfect time division multiplexing by allocating a virtual transmission time for each packet given by the rate allocated to its flow. On the arrival of a packets, its virtual transmission time is set to the previous packet time plus a transmission time given by the ratio between the packet length and the allocated rate. Packets are then sent in the order of their transmission times. The problem with this computation is that a flow that remains idle accumulates priority for future transmissions. It was proposed to change the previous packet time in the equation by the maximum between previous packet time itself and the arrival time of the new packet. However, it may penalize excess usage even if most of the flows are not incorporating new packets.

Scheduling Protocols attempting to achieve delay differentiation Following the framework introduced in [111], the Proportional Differentiated Services (PDS) framework, many methods have being proposed to achieve delay differentiation. The idea is to provide a differentiation between flows or queues by trying to have a ratio of average delays equal to the ratio of differentiation parameters. **A scheduler from this framework has been taken as the last option for our study**.

RESTRICTED DOCUMENT

[112] extends WFQ to achieve service differentiation by adjusting the weight of each class according to the arrival rate and buffer occupancy in order to control the delay differences. They achieve the desired differentiation when using a short weight adjustment period of 0.25 seconds, which poses a problem from the complexity side.

[113] proposes a new protocol, Scaled Time Priority (STP), attempting to reduce the complexity of previous schedulers. The computation of the priority of a queue is only calculated when a new packet reaches the head of the queue. STP approaches well to the behavior of the scheduler presented in [111], which in its turn has some difficulties meeting the differentiation depending on the load and characteristics of the traffic.

A modification of DRR called Efficient and Robust Dynamic Deficit Round-Robin (ERP-DDRR) was introduced in [114]. It does not require timestamping or monitoring of the packet arrival. In addition to the different FIFO queues, they introduce token queues and allocate the number of token queues for each class according to packet priority. The higher priority tokens can be distributed to a larger number of queues increasing the probability of being scheduled. As the number of token queues is increased, the achieved differentiation approaches the given targets.

Finally, a scheduler with buffer management was proposed in [115]. Fair Bandwidth Sharing with Delay Differentiation integrates packet scheduling with buffer management, trying to control loss rate differentiation in the process as well. They propose two schedulers that compute the normalized throughput of each class given by the ratio between the waiting time and the size of the packet multiplied by the differentiation parameter. Then the buffer management removes packets from the tail of the queues when there is an overflow and tries to ensure a relative loss rate between the services given by the loss differentiation parameter. Overall, the framework is very similar to the one we have taken from PDS, but it allows better differentiation when the average packet size from one service to another differ, which is not the case for the scenario we consider.

7.2 Proposed Scheduling Protocols

7.2.1 Adaptive Deficit Weighted Round Robin (DWRR)

DWRR [116] divides traffic in N classes and gives each class a quantum that represents the total number of bytes that a class should send at each round. In the case some bytes are not used, they are reported to the next round. Consequently, each class is given a share or weight of the output capacity that equals to the ratio between the quantum of the class and the sum of all quantum's.

In our scheduler, frames are classified between VoIP and PTT and are put in separated FIFO droptail queues. It is possible to achieve the desired delay differentiation and limit maximum queuing delay by correctly selecting the weight and the size of each queue. The implementation complexity of this scheduler is light and configuration of parameters is the main challenge. Figure 7.1 provides an overview of the discussed protocol.

RESTRICTED DOCUMENT

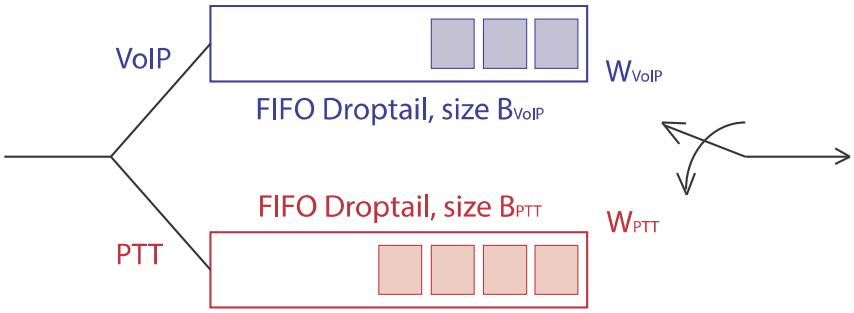


Figure 7.1: Overview of the DWRR Scheduler

7.2.2 Jitter/Delay Earliest Due Date (EDD)

The second solution we present is an adaptation of a timestamped scheduler that assigns the transmission priority based on deadlines. Frames are no longer divided between VoIP and PTT. Instead, a single queue, ordered by deadline, is used. Frames with earlier deadlines will be located at the beginning of the queue and, hence, they will be served first.

Delay EDD [117] imposes a deadline or a delay bound to a packet in each router. Upon the arrival of a packet, the router determines the arrival time and calculates the deadline by adding the local delay bound. A packet is then dropped if it cannot be sent before its deadline. Jitter EDD is an evolution of the described algorithm that tries to limit jitter by imposing an extra waiting time to the packets that were transmitted before its deadline in the previous router.

In our adaptation, the protocol is just implemented in the MUX. In order to calculate the deadline we determine the difference between the actual delay and the typical one prior the arrival to the MUX before adding the delay bound. This technique limits the effects of the jitter caused by retransmissions within the LTE network. The frames that had to be retransmitted will have a lower queuing delay. In order to perform the service differentiation, two different delay bounds are imposed.

After a MUX packet is sent, the protocol removes the frames inside the queue which were unable to meet their deadline. Such policy enforces a strict delay bound. The processing load increases due to the need to systematically re-order the frames in queue, but configuration is straightforward. Figure 7.2 summarizes the described mechanism.

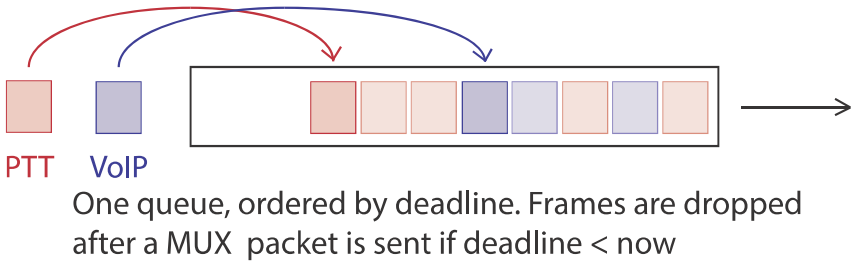


Figure 7.2: Overview of the Jitter/Delay EDD Scheduler

RESTRICTED DOCUMENT

7.2.3 Hybrid Proportional Delays (HPD)

The last scheduler was proposed as part of the "Proportional Differentiated Services" framework [111]. The authors suggest a series of schedulers to perform delay service differentiation. Their objective is to perform a continuous relative differentiation, by trying to have a ratio of average delays equal to the ratio of differentiation parameters. Differentiation is only feasible when input traffic durably exceeds the transmission capacity and some frames must be delayed. Mathematically, their goal can be expressed as follows:

$$\frac{d_i}{d_j} = \frac{\delta_i}{\delta_j} \tag{7.1}$$

Where d_i, d_j are the average delays of classes i and j; δ_i and δ_j are the differentiation parameters. Higher classes receive a service that is proportionally better than those of lower classes. The first class, considered the worst, has a differentiation parameter of reference equal to 1.

$$d_i = \delta_i d_1 \tag{7.2}$$

$$\delta_1 > \delta_2 > \ldots > \delta_N > 0 \tag{7.3}$$

We consider their last proposed scheduler, the hybrid proportional delay, which aims at having a proportional delay differentiation in the long term, in average, and in the short term, instantly.

In our scenario, traffic is divided in two FIFO queues. At every transmission opportunity, the protocol selects the service with the current maximum "normalized hybrid delay". This parameter consists in a component of average delay and another of instant queuing delay of the head of the queue:

$$h_i(t) = g\bar{d}_i(t) + (1-g)\overline{w}_i(t) \tag{7.4}$$

Normalized average delay:

$$\bar{d}_i(t) = \frac{d_i(t)}{\delta_i} \tag{7.5}$$

Normalized head waiting time:

$$\overline{w}_i(t) = \frac{w_i(t)}{\delta_i} \tag{7.6}$$

Where the parameter g $(0 < g < 1)$ gives a relative weight to each component. The protocol tries to maintain a constant ratio of the delays of the different services in the short and long term. The algorithm chooses the service that, at a given time, is further from its objective.

When the system is overcharged, some frames must be dropped. For that we follow the second part of the framework [118] that aims at providing loss rate differentiation. However, we enforce a ratio, equal to one, giving the same dropping rate to both services. Instead of having a queue size for each queue, there is a unique global size. When the queue is full, a frame from the service with the current lower dropping rate is dropped. In this way, the purpose is to equalize the dropping rate of both services.

RESTRICTED DOCUMENT

This global buffer size shall be well configured to meet the maximum latency goals, as the protocol itself just performs a relative differentiation.

The complexity for this scheduler is high. Frames must be timestamped and the average queuing delay is constantly updated. Additionally, the dropping rate has to be calculated as well at each dropping decision. Figure 7.3 shows an overview of the HPD scheduler.

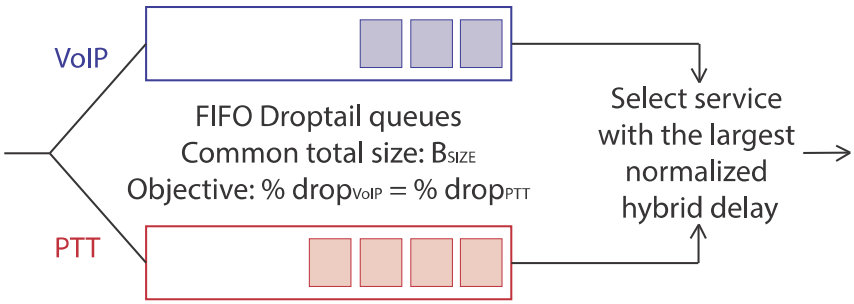


Figure 7.3: Overview of the HPD Scheduler

7.3 Simulation and Performance Evaluation

The performance evaluation of the different scheduling proposals is based on the same scenario and parameters we observed in the previous chapter. The multiplexer was modified to implement the scheduler of choice instead of having a FIFO queue prior the mux packet creation. Figure 7.4 recalls the scenario previously taken. The QoS requirements were also the same and the first objective was to derive the resulting capacity gain thanks to differentiation between VoIP and PTT.

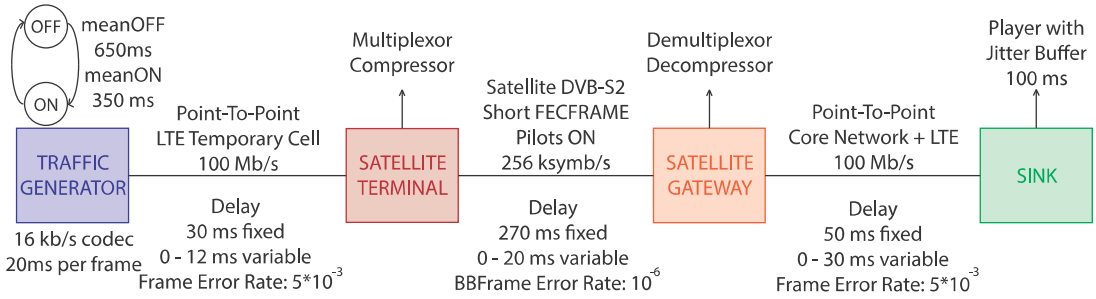


Figure 7.4: Simulated Scenario

7.3.1 Results of Service Differentiation Proposals

This subsection concerns the comparison of the service differentiation scheduling possibilities. The objective was to impose a delay differentiation ratio of 3, taking the queuing delay of a DVB-S2 frame duration as the maximum for VoIP. We relaxed this constrain for the case using 32APSK 4/5 increasing

RESTRICTED DOCUMENT

this to two, taking advantage of the reduced frame duration. Given the selected symbol rate, for the same ModCod as before, these delays were 32, 21, 26$(2*13)$ milliseconds approximately.

First, we wanted to analyze the effectiveness of the scheduling differentiation by measuring the derived maximum capacity. Figure 7.5 shows the number of gained users for three different loads of VoIP and PTT sessions using the average results of the presented options. Generally, we obtained the same results for the different solutions, yet, in most of the cases, DWRR supported one user less than its competitors. We see that in the case with QPSK the gain is quite remarkable but this is also because we allowed more queuing delay in absolute. Similarly, the case with higher spectral efficiency yielded a better result because we doubled the value of accepted PLFRAME duration delay.

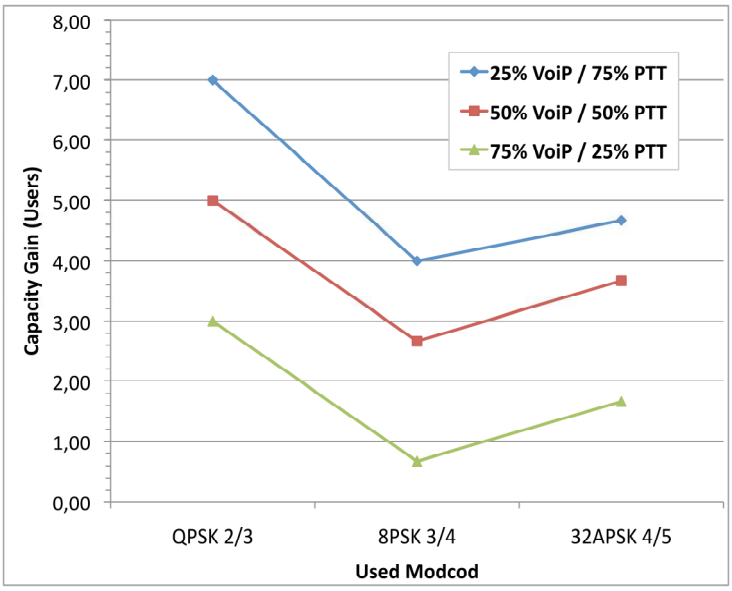


Figure 7.5: Maximum Capacity Gain using Differentiated Scheduling

7.3.2 Delay Comparison of Service Differentiation Proposals

It is interesting to observe the delay comparison with the case without differentiation in figure 7.6. The average delay of VoIP is practically equal and the maximum is lowered. Comparing the different scheduling proposals, it was difficult to adequately parameterize some of the options to limit the maximum queuing delay. We observed a lot of difference with the 99th percentile, a value that was simpler to measure and limit. The average PTT delay did not increase excessively though.

Figure 7.7 and 7.8 compare the proposals in terms of queuing delay for VoIP and PTT respectively. The case with 50/50 load was taken given that we did not observe a lot of difference, except for the average delay of PTT, which increased slightly with the ratio of PTT flows in the mix. The EDD case was the only one that could effectively impose a determined maximum value, the 99th percentile had to be taken for the rest of options as benchmark. We omitted the average delay because the results were similar to the values showed in figure 7.6.

RESTRICTED DOCUMENT

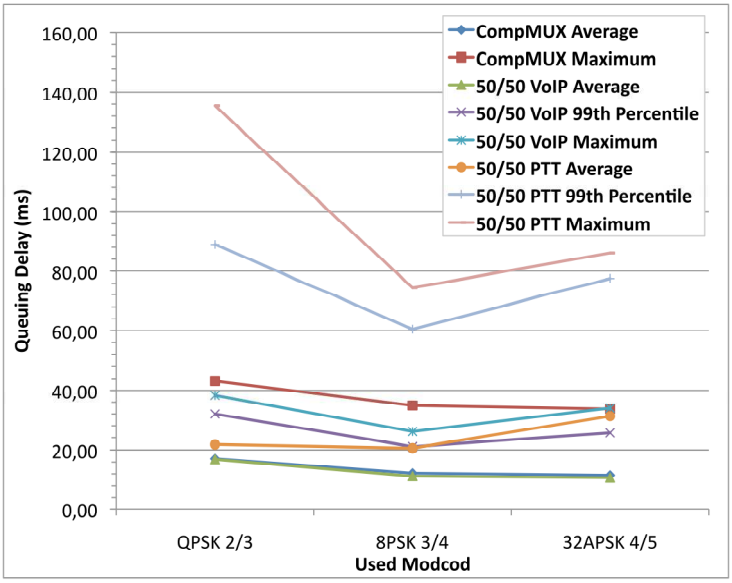


Figure 7.6: Delay Comparison

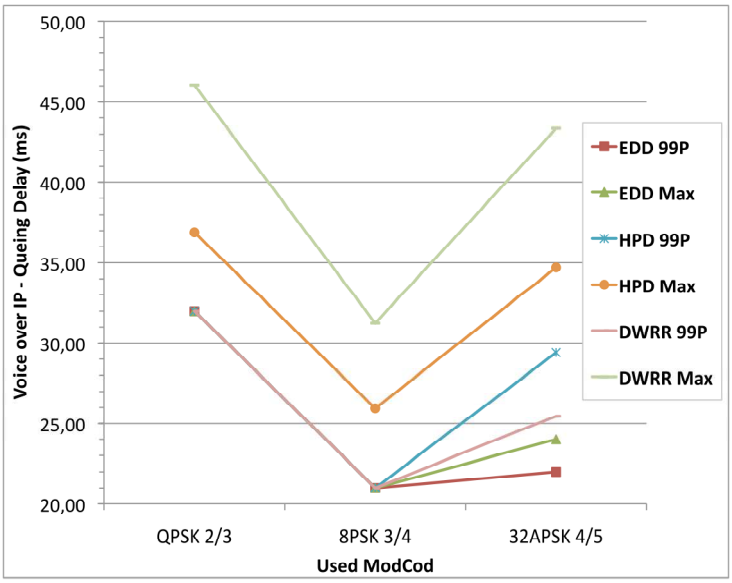


Figure 7.7: VoIP Queuing delay comparison

Figure 7.9 derives the delay differentiation parameter given by the ratio of PTT and VoIP resulting delays. Again, considering the maximum for EDD and the 99th percentile for the rest, all were close to the desired mark of three. Although HPD controls the delay differentiation parameter at all times, it was not able to meet its target for the average delay metric because of the PLFRAME duration restriction and the transmission of voice frames at the same time inside a MUX packet.

RESTRICTED DOCUMENT

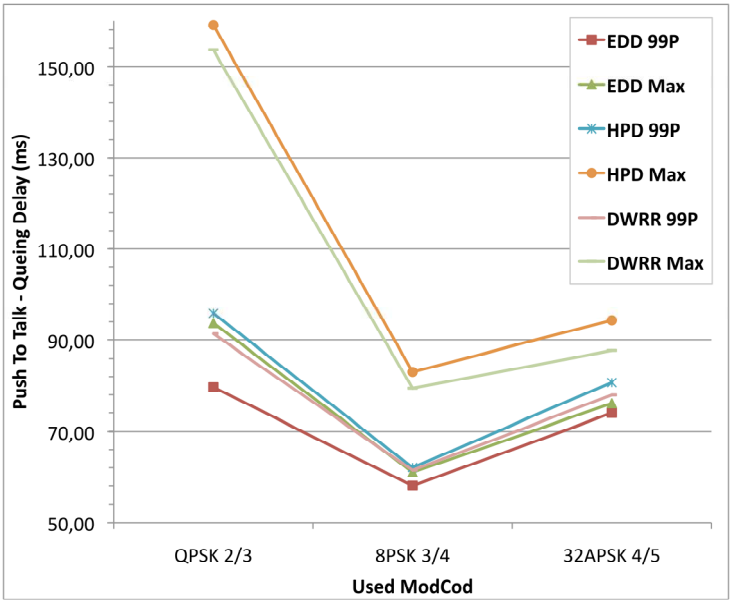


Figure 7.8: PTT Queuing delay comparison

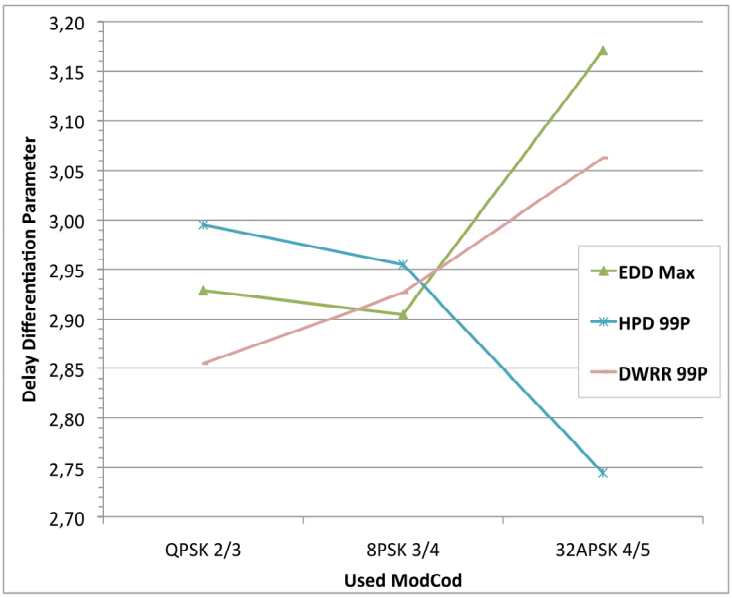


Figure 7.9: Delay Differentiation Parameter

7.3.3 Drop Comparison of Service Differentiation Proposals

Finally, a practically equal dropping rate for the two services should be guaranteed. Figure 7.10 shows the results for the drop differentiation parameter, the ratio between the dropping rate of PTT and VoIP, where the objective was to get close to one. HPD was explicitly designed to do so and performed perfectly. DWRR was able to get close after well tuning the parameters. However, EDD was far from the goal, as it has absolutely no control over which service should be dropped.

RESTRICTED DOCUMENT

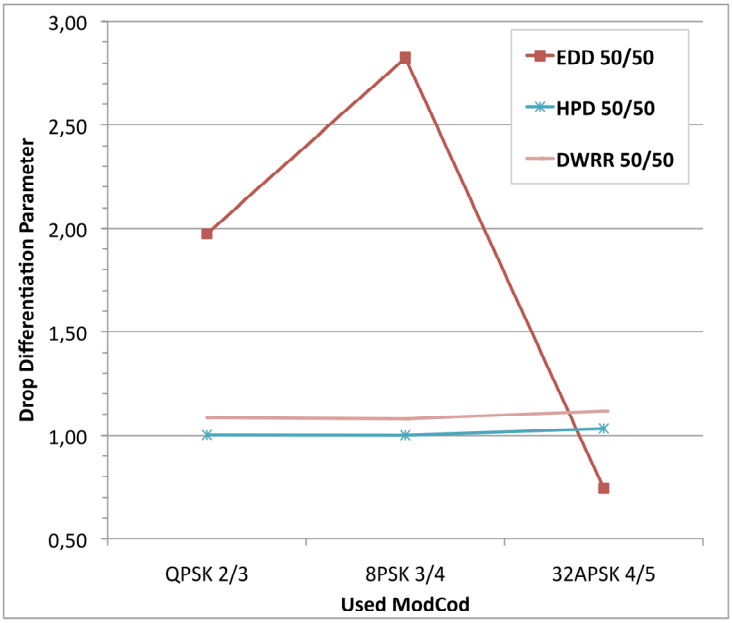


Figure 7.10: Drop Differentiation Parameter

7.3.4 Conclusions

We proposed to provide service differentiation between VoIP and PTT in order to push the capacity further. We presented three different options: the first alternative, DWRR, approached well the objectives but was very complicated to tune, unmanageable for a real-time system; HPD was better but increased notably the necessary processing power, which was between 3 and 4 times higher than the other two cases; finally, EDD was the only one that perfectly limited the maximum delay but could not manage to balance the dropping rate of both services. On the good side, EDD just needed the queuing delay imposed deadlines.

RESTRICTED DOCUMENT

Chapter 8

Adaptation to traffic changes

8.1 Introduction

The idea of this last chapter is to integrate the satellite access technique based on rate demand and random access (RBDC+RA) with the jitter Earliest Due Date Scheduler presented in the previous chapter. We simulated PTT groups and VoIP calls coming from the satellite LTE cell and therefore using the satellite return link.

For the PTT side, we have considered that the user is transmitting at all moment during the message, i.e. without voice activity detection. Additionally, some of the calls can be coming from the other side of the satellite, which actually reduces the load on the return side. For VoIP, the classical voice activity model has been taken, similar to the previous chapters. When there is a new call starting, the satellite terminal issues a demand for the total capacity of the call, not considering the possibility of benefiting from the extra capacity available from other calls when the users are not speaking.

Considering system parameters similar to the ones taken in the satellite access techniques comparison, we observe that the system works well in general. However, there are two main issues that make the solution stumble: (a) the overload, given that without call admission control all calls/messages are accepted, and (b) the sudden arrival of multiple calls, increasing the load during a period of time before updating the resource demand. This chapter concentrates in the latter. A large load coming from VoIP calls does reduce the issue given that normally the system will have some spare capacity. Hence, we focus on the cases where the load coming from PTT messages is more important.

The rest of this chapter is organized as follows: section 2 presents the problems we may encounter in a scenario like the one presented above. Then, section 3 proposes two possible solutions and shows the results of some specific cases. Further, the following section evaluates the proposed solutions in further simulations, not focusing on a given case but observing how the solutions compare dealing with the arrival of multiple calls in a short period of time. Finally, we provide our conclusions for the chapter.

8.2 Problems identified

As it has been stated, by taking advantage of the random access mechanism, the system is usually well prepared to deal with the arrival of new calls while keeping the delay low. However, we try to limit the access through random access because RA is not efficient when dealing with higher loads. Additionally, there exists is the possibility of having a full allocation, making it impossible to transmit in the random access channel.

The potential issues we have identified occur when multiple calls arrive into the system in a short period of time. In our example, there is a resource demand opportunity each 270 *ms* (one-way delay of the satellite link), which results in an update delay of one round-trip delay. The allocation delay of new resources may imply that during some periods the system is overcharged.

This situation may provoke multiple packet drops at the beginning of new calls that could prevent the listener to understand the start of the message. These bursts of errors are what diminish most the quality of experience, specially when they occur at the beginning of the session. Furthermore, we do not consider here other causes of error, such as those corresponding to the mobile network physical layer (in the transmission or reception side) or on the satellite link, which generally occur in bursts as well. Therefore, our interest here is to reduce the situation where multiple packets are dropped due to excessive delay.

Next, we review some examples of these situations and observe how they translate into the delay and drop planes. Two types of graphs will be presented, the ones regarding the events (call arrivals and departures, packet drops...) and the ones showing the queuing delay.

In the first ones, we can observe the arrival of the PTT calls into the system and drop events that may occur. The Y-axis shows the number of calls that are currently managed by the satellite terminal for the blue line. In red we observe the number of PTT packets that are dropped in a given instant because they achieve the maximum waiting time on queue (60 *ms* for PTT packets). More than one packet can be discarded at a time. In green we see the instants where a demand frame is sent to the network controller. The demand will consider the number of calls that are currently into the system. In this case the Y-axis is not important, because we only care about the moments, the X-axis, which is the simulation time.

The second ones show the evolution of the queuing delay of the sent packets in blue. We decided to also include the dropped instants in order to show their direct correlation with the augmentation of the queuing delay.

The first example shows the case of a cold start. At a given instant, there are no users transmitting and multiple calls start arriving. The system is using the random access mechanism to send the first packets but as we limited the number of frames the terminal can transmit using this option, some errors are originated. In figure 8.1 we see up to three arrivals around 73.5 seconds, some packets need to be dropped before allocation for at least the first call occurs. Figure 8.2 shows how the queuing delay increases during this period, before returning to normal levels.

The second example is the case where the system sees the end of multiple calls and, just when the

RESTRICTED DOCUMENT

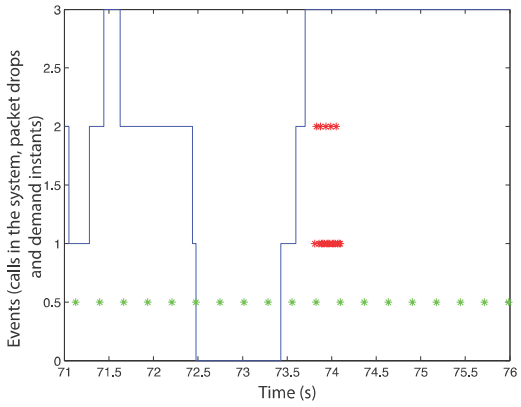


Figure 8.1: Events. Example 1

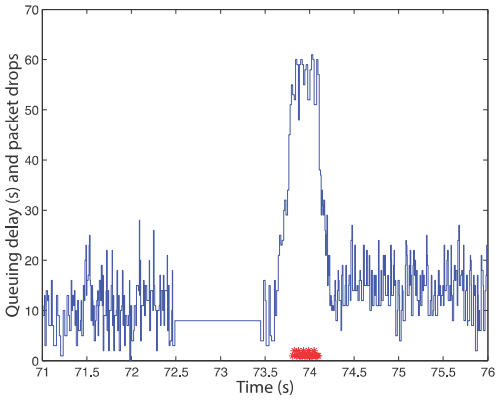


Figure 8.2: Queuing Delay. Example 1

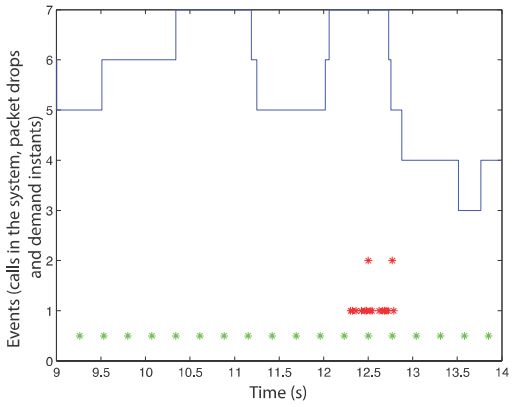


Figure 8.3: Events. Example 2

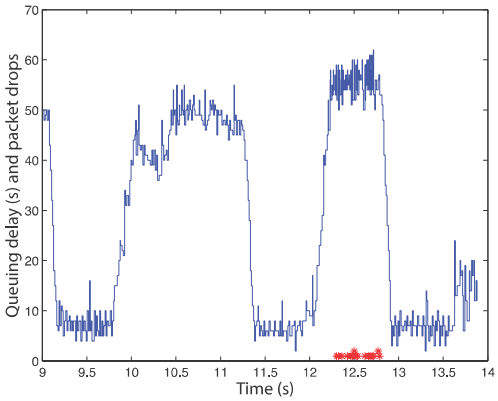


Figure 8.4: Queuing Delay. Example 2

allocation has been updated, an equal or greater number of calls are initiated. In 8.3 we observe this phenomenon between 11 and 12 seconds. The system is overloaded during less than a second, until some calls terminate. Figure 8.4 shows the queuing delay. We see that previously there has been a period of overload, but thanks to the delay adaptability it has been well resolved.

In the third example we see the arrival of multiple calls in short time around the 47 second of figure 8.5. The delay clearly increases during the allocation period and multiple packets are discarded. We see that close to the 45 second a similar situation occurs but in a lighter way. Figure 8.6 allows us comparing these two events in the queuing delay part.

The forth example has clearly two parts, observed in figure 8.7. In the first one, around 6.5 and 7 seconds, we see the arrival of new calls that create some overload tension. The second part is not that important for our study. We achieve the maximum capacity, using all the possible frequency band, around 7 calls. Therefore accepting an eighth one creates a continuous overload. In this study we decided not to use a call admission mechanism, so we allow this to happen. However, it is a dimensioning issue, not a quality of service one. Hence, we will not be able to do much about it. Figure 8.8 presents the queuing

RESTRICTED DOCUMENT

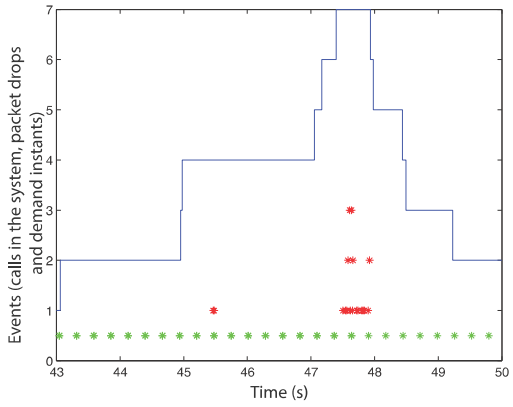


Figure 8.5: Events. Example 3

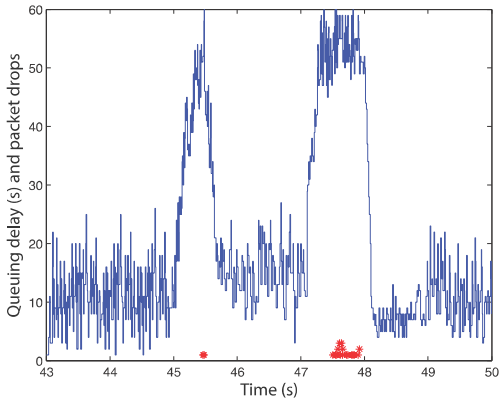


Figure 8.6: Queuing Delay. Example 3

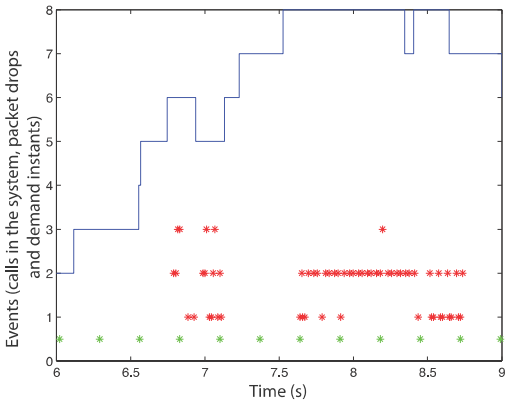


Figure 8.7: Events. Example 4

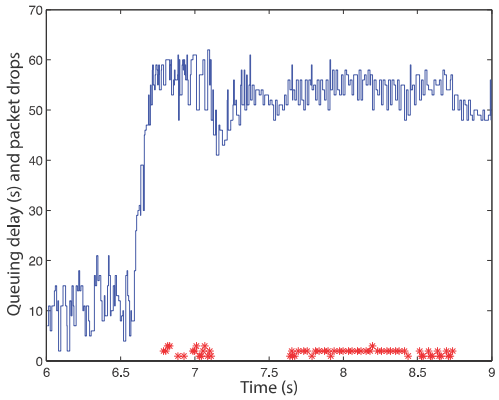


Figure 8.8: Queuing Delay. Example 4

delay, where the zone of load, with a queuing delay of $50 - 60$ *ms*, is present after the first commented part.

Finally, the last example also shows the behavior when the number of calls overpasses the maximum capacity. Yet, here some calls are from the VoIP side (presented in the negative side of the graph for simplicity). As not all VoIP are active at the same time, the system manages somehow better the situation. The cyan line in figure 8.9 shows the number of VoIP calls in active state and therefore transmitting. The magenta points are the dropped packets for the VoIP flows, which have been limited to a 25 *ms* queuing delay. Figure 8.10 shows the delay. Again, in the negative side we find the VoIP flows, in green. We clearly see the period of overload where the delay increases until the maximum allowed for both services and multiple packets from each side are dropped.

Table 8.1 summarizes what it has been observed in the commented scenarios and the causes.

RESTRICTED DOCUMENT

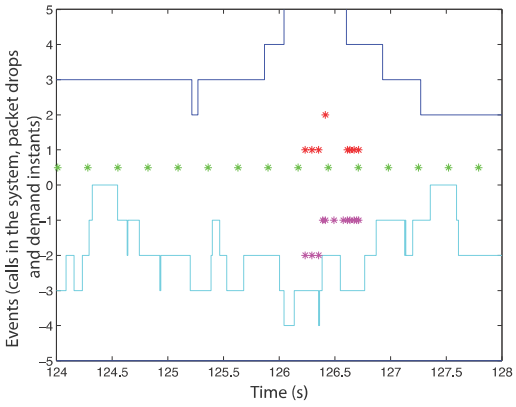


Figure 8.9: Events. Example 5

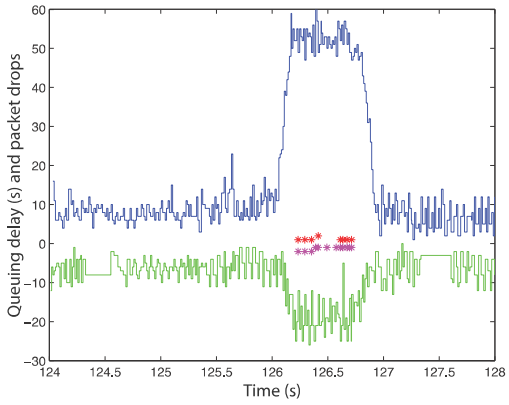


Figure 8.10: Queuing Delay. Example 5

8.3 Proposed Solutions

In order to solve the problems identified in the previous section we propose two different solutions called ”degraded mode” and ”start-up delay”. Both solutions play with the end-to-end delay of the PTT messages in order to reduce the issue of having multiple calls arriving in a short period of time.

The first task, for both solutions, is to identify the calls that may cause a burst of errors. The availability of a random access method allows the system to deal with overloads of least magnitude caused by the arrival of new calls. However, we propose to apply the solutions in case the system detects that we have an overload of two calls with respect to the current allocation. One shall notice that the current allocation may differ from the one in the following superframe, given to the traffic variation. Hence, it is also of interest to observe if we will be in overcharge after the acceptance of the new message in the near future, before the actual capacity update.

Next, we present the possible solutions and observe how they work in some of the cases we have identified in the previous section.

8.3.1 Degraded Mode

Following the principle of the selected scheduler, we propose to apply a degraded mode for the calls that have just arrived to the system and may pose an issue due to a provisional and temporary overload. Generally, we consider two types of traffic, VoIP and PTT, and assign a deadline to each packet that is a direct function of the maximum waiting delay allowed for each class. For this solution we add a new class, the degraded PTT, which implies a much higher maximum waiting delay for the packets of the calls that have just been admitted and that challenge the current allocation. This solution directly accepts new calls and, while giving them a lower priority, allows the use of all available transmission opportunities. This is a work-conserving solution.

The adoption of this degraded mode translates to a higher queuing delay at the beginning of the calls and likely a larger jitter. Therefore, it is necessary to increase the jitter buffer in the reception side in

RESTRICTED DOCUMENT

Example	Observation	Cause
1	Short delay increase and several packets being dropped	Cold start: arrival of multiple calls when the system had no allocation
2	Two periods of delay increment. Only the second has the consequence of packet discard	Arrival of two calls just after the allocated rate was reduced
3	Two short intervals of delay rise, with drops at the end of which	Two cases of multiple call initiation in less than 270 *ms*
4	A period of delay increase and packet dropped that is maintained during some seconds	First, the arrival of two calls consecutively and then, the maximum capacity is overpassed
5	A rise in the queuing delay of PTT and VoIP resulting in numerous discarded packets from both services	In presence of five VoIP calls, the arrival of two additional PTT messages exceeds the capacity of the system

Table 8.1: Summary of the presented examples

order to guarantee that the message is delivered without noticeable interruptions.

We propose to leave this job for the receiving Super Node, which will have to be informed that the given call is in transmitted in degraded mode. A good synchronization between the network control center (NCC) at the gateway side and the SN would avoid the explicit need of such notification. This approach will leave all the work to the SNs and let users operate as usual.

In the simulated system, we considered 25 *ms* for the maximum delay of VoIP, 60 *ms* for regular PTT calls and 200 *ms* for the degraded PTT calls. This three class model translates to a prioritization of the PTT messages already in the system and increasing notably the delay on the new ones. Next we review the behavior of this solution in the examples presented in the previous section.

The first example graphs are displayed in figures 8.11 and 8.12. We observe that it does reduce the number of packets dropped but it does not resolve totally the issue. In the queuing delay figure we include in black the delay for the flows in degraded mode. We see how the delay for these flows increases steeply first, but the one for the first accepted call does increase as well and, being all close to their maximum allowed delay, some packets are discarded.

In the second example, figures 8.13 and 8.14, we observe a similar behavior. The effects of the overload are somehow reduced but not eliminated. In figure 8.14 one can observe the variability of the delay in the flows that have been marked to be managed in degraded mode. During load phases their delay increases rapidly to the maximum levels and in light periods they tend to behave as the other flows.

Figures 8.15 and 8.16 present the third example with the degraded mode solution. The arrival of two consecutive calls around 45 seconds increases the queuing delay and we see that some packets are discarded. In the following call arrivals the problem is reduced but not eliminated. It is expected that

RESTRICTED DOCUMENT

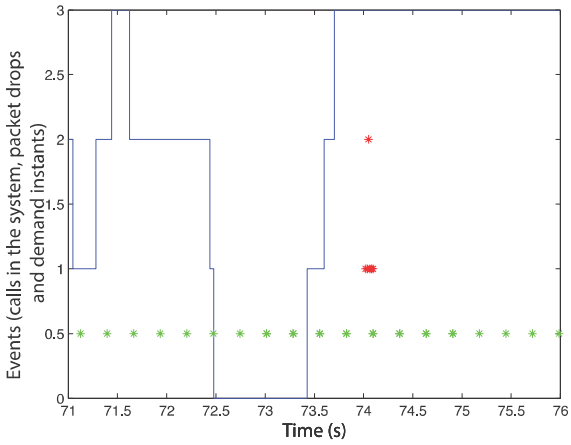


Figure 8.11: Events. Ex 1. Degraded Mode

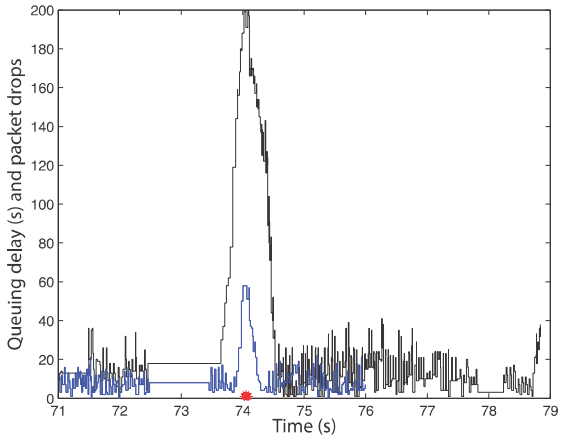


Figure 8.12: Delay. Ex. 1. Degraded Mode

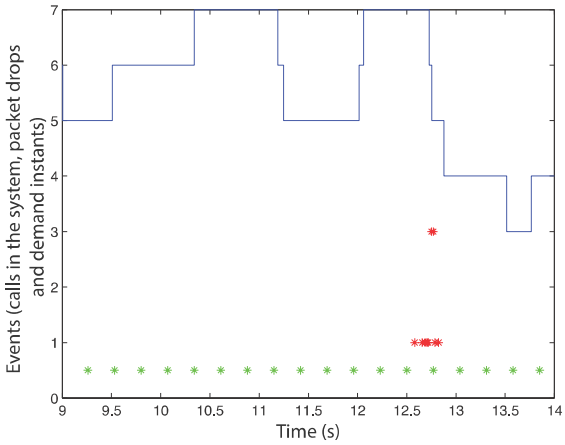


Figure 8.13: Events. Ex. 2. Degraded Mode

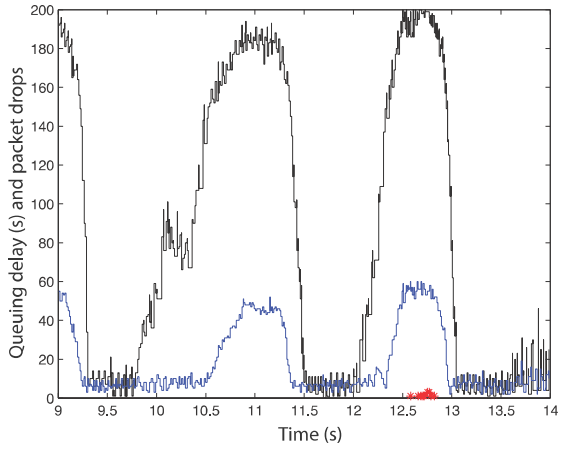


Figure 8.14: Delay. Ex. 2. Degraded Mode

the number of packets dropped tends to be distributed among all PTT calls, reducing the effects on the quality of experience.

In the forth example, figures 8.17 and 8.18, we can see that the usage of this prevention method cannot effectively manage a situation overpassing the maximum possible capacity. Yet, in the first part of the example, the number of packets discarded is reduced. We see in the delay figure that the delay levels are close to the maximum values since the acceptance of the calls of the first part. The situation will take a bit to stabilize when the number of calls is reduced to manageable levels.

Finally, while the last example showed a case dealing with maximum capacity, the situation was managed thanks to the degraded mode. As we observe in figure 8.19, no packets were dropped. In figure 8.20 we clearly see how the prioritization of services works. First the calls in degraded mode start seeing their queuing delay increased, then the regular PTT calls and finally the VoIP flows. Once a call terminates, around 126.5 seconds, the situations returns quickly to normal.

RESTRICTED DOCUMENT

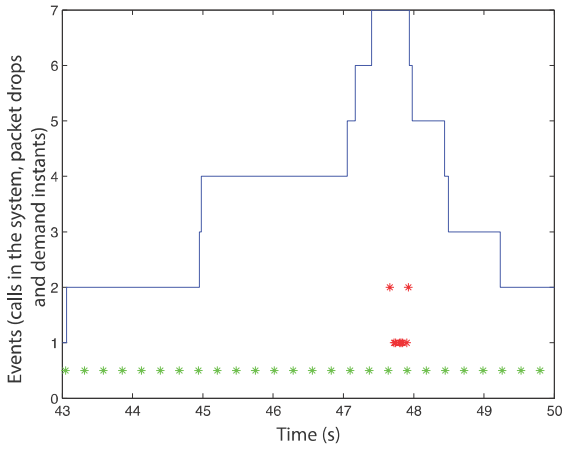


Figure 8.15: Events. Ex. 3. Degraded Mode

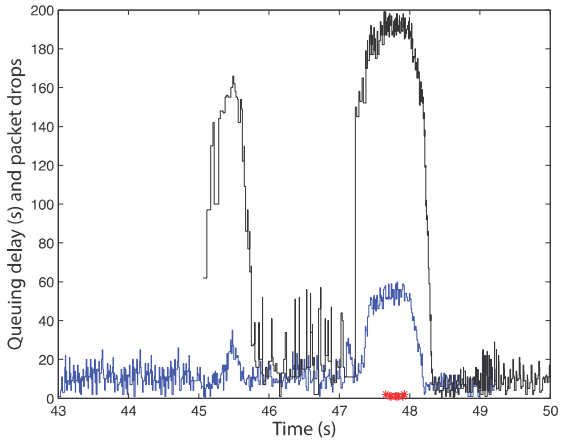


Figure 8.16: Delay. Ex. 3. Degraded Mode

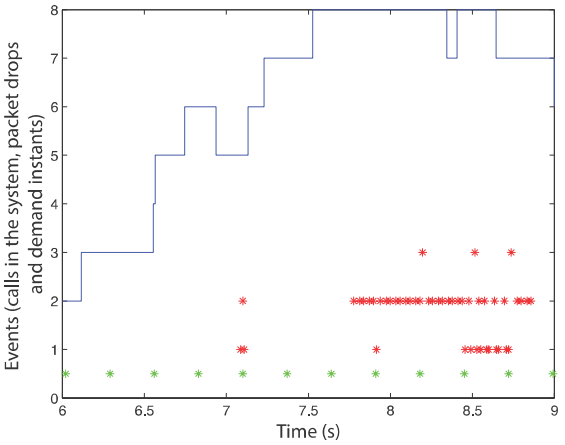


Figure 8.17: Events. Ex. 4. Degraded Mode

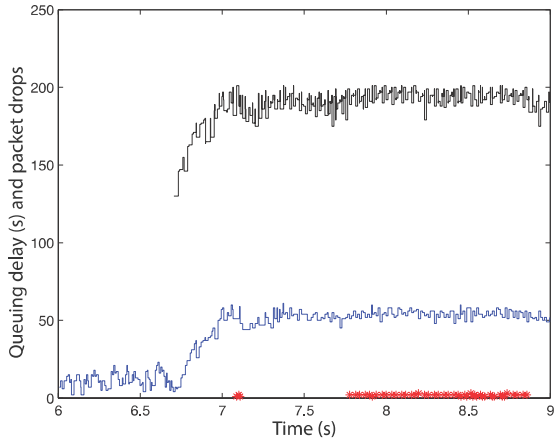


Figure 8.18: Delay. Ex. 4. Degraded Mode

8.3.2 Start-up Delay

The second solution is to actually delay the start of a new PTT call in case it is detected that it can produce a burst of errors. It is a non-conservative solution, some transmission frames can be left unused if there is temporarily no packets to transmit and the new call has not yet started being transmitted. Yet, it provides a mode adaptable solution in the sense that there is not a given start-up delay but that it is chosen depending on the load and the instant when the request for the new call is received. After this start-up delay, the message is processed as any other PTT message with the same priority in terms of delay.

The system will delay the call until the start of the next SuperFrame or the following one. It has been noticed that the system can manage well an overallocation of two calls during some period, given the deadline they have been given. The worst case occurs when the system is detected to be in overload for

RESTRICTED DOCUMENT

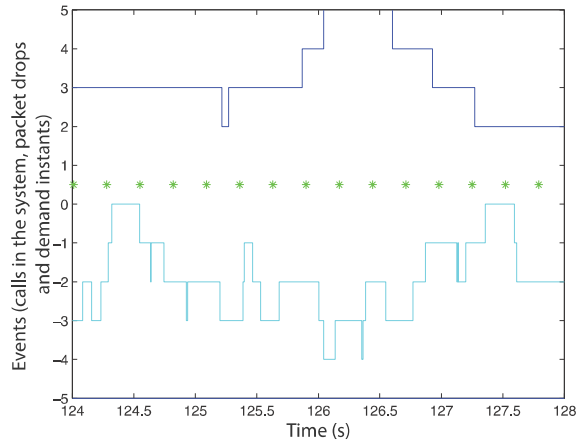


Figure 8.19: Events. Ex. 5. Degraded Mode

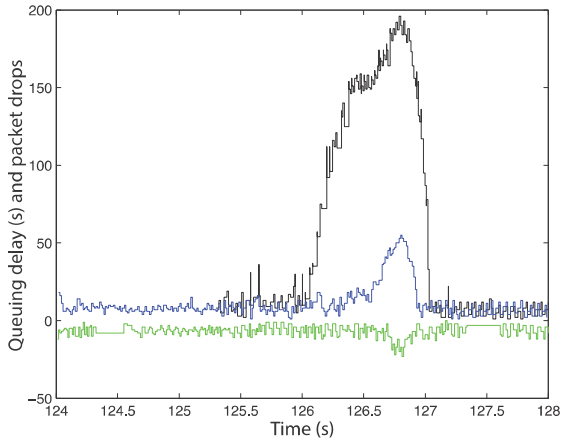


Figure 8.20: Delay. Ex.5. Degraded Mode

a long period and that the call shall be delayed until the allocation process is updated taking this call into account. This adaptation yields a very variable start-up delay for the concerned calls, which can be just 50 *ms* or around 700 *ms* for the worst cases. The review of the previously discussed examples allows to extract some conclusions regarding this issue.

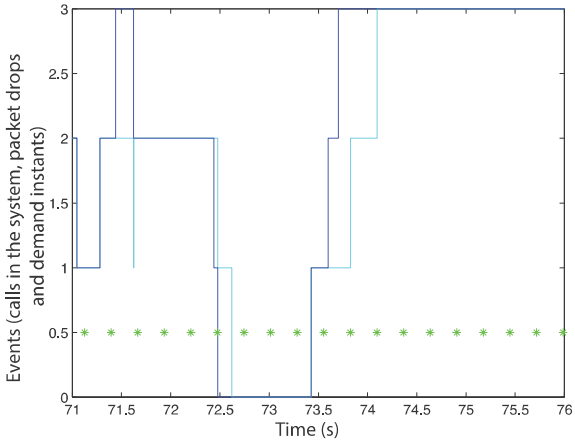


Figure 8.21: Events. Ex. 1. Start Delay Mode

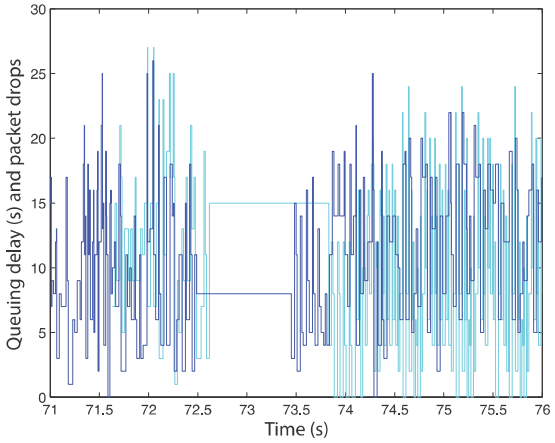


Figure 8.22: Delay. Ex. 1. Start Delay Mode

The situation in the first example is perfectly resolved with this approach. By delaying the start of the conflicting calls, we avoid all packet discard. The cyan line in the events figure, 8.21, shows the evolution of the number of calls that are currently being transmitted. One can easily see how some calls are delayed. Between 71.5 and 72 seconds we observe how a call that ends is just replaced by another one that was waiting to be started. In figure 8.22, the cyan line shows the queuing delay for the calls whose beginning has been delayed. We do not appreciate any difference with the other PTT flows once the calls are started.

RESTRICTED DOCUMENT

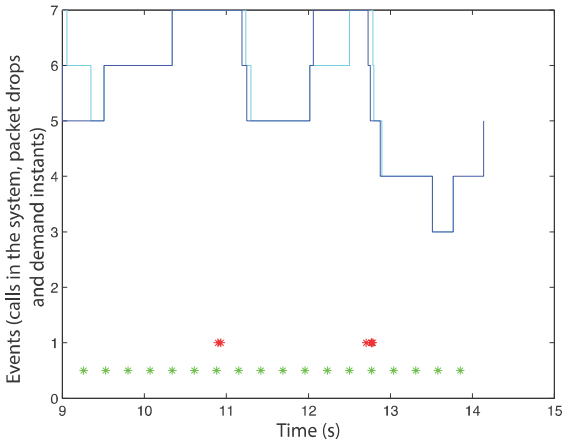


Figure 8.23: Events. Ex. 2. Start Delay Mode

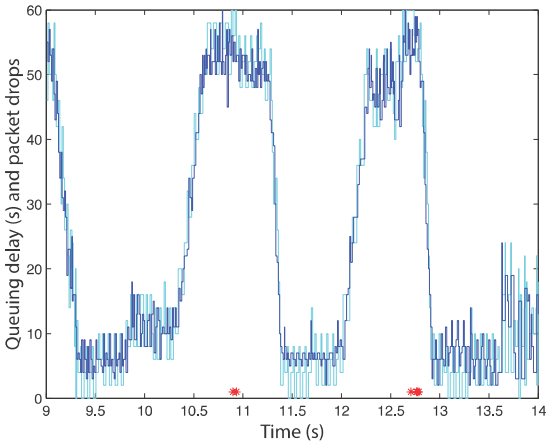


Figure 8.24: Delay. Ex. 2. Start Delay Mode

In the second example we see how the method alleviates the overload in the around the 12th and 13th seconds of figure 8.23 but other calls that were delayed previously generate some losses around the 11th second. In figure 8.24, we observe this two load periods with the consequence of some losses. Given that multiple calls are being transmitted, this situation usually translates to one or two losses in a short period, which can be handled by the loss concealment techniques in the receiver terminals.

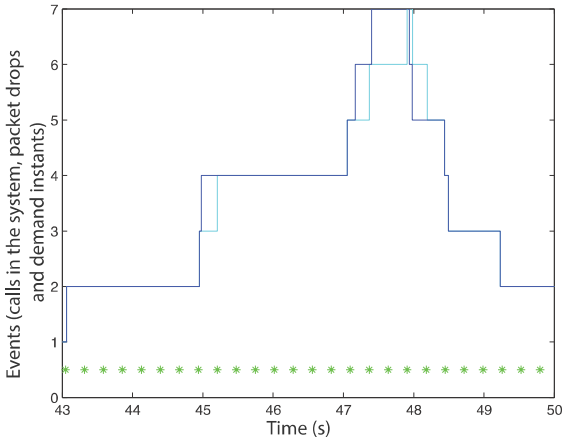


Figure 8.25: Events. Ex. 3. Start Delay Mode

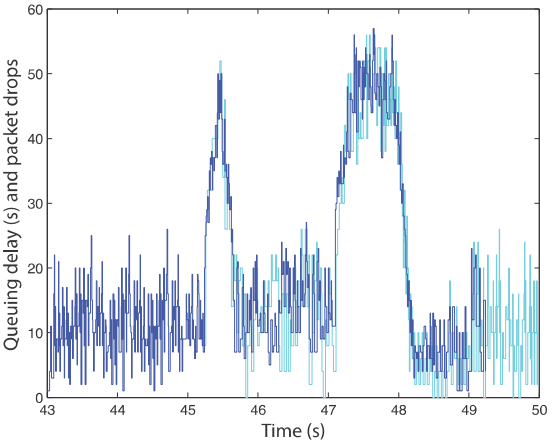


Figure 8.26: Delay. Ex. 3. Start Delay Mode

In contrast to the previous method, the adoption of a preventive start-up delay does solve the situation of the third example in figures 8.25 and 8.26. In the events figure we see that the time transmitting 7 calls is considerably reduced. This may introduce some period of slight inefficiency because the terminal does update the demand for new resources when the calls arrive, even if a situation of overallocation may possibly occur later.

In the forth example, figures 8.27 and 8.28, we see a similar behavior to the previous case with

RESTRICTED DOCUMENT

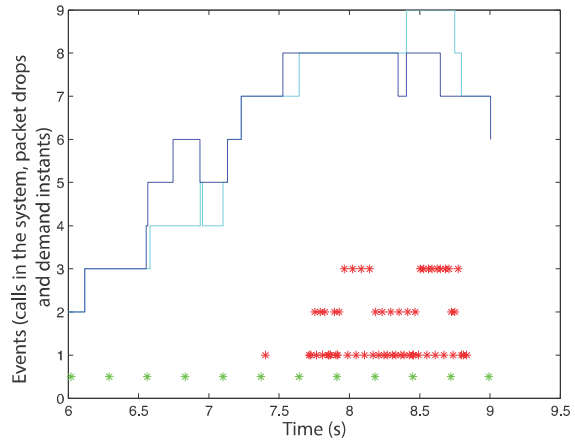


Figure 8.27: Events. Ex. 4. Start Delay Mode

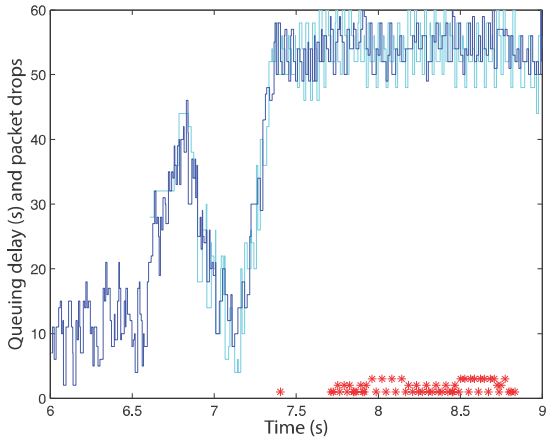


Figure 8.28: Delay. Ex. 4. Start Delay Mode

degraded mode. The first part is well handled, eliminating all packet discard. However, the approach does not delay the start of a message indefinitely while we are overpassing the capacity. Therefore, most of the losses resulting from this will not be eliminated. Indeed, we observe that because some calls where delayed, we even have a period of time with a larger number of calls being transmitted compared to the case without prevention.

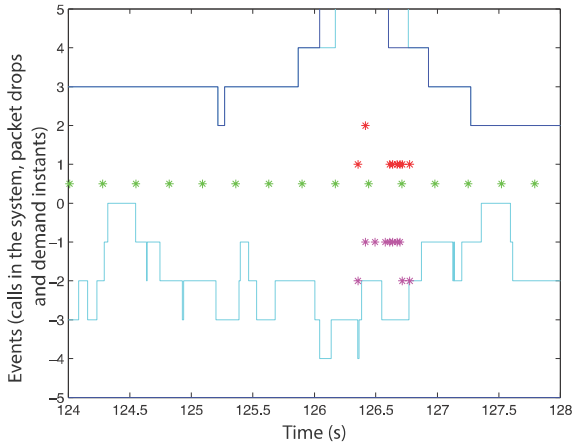


Figure 8.29: Events. Ex. 5. Start Delay Mode

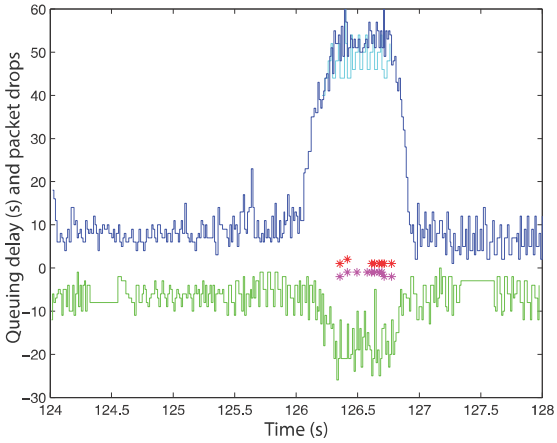


Figure 8.30: Delay. Ex. 2. Start Delay Mode

Finally, we see that this method does not avoid the period of excessive traffic in the last example. In figure 8.29 we see that just one call is actually delayed and figure 8.30 reminds us the figure showed at the beginning. We observe a period of overload around 126.5 seconds that results in many packets discarded from both services, PTT and VoIP.

RESTRICTED DOCUMENT

8.4 Evaluation

This last section compares the outcomes of the two prevention methods presented against the baseline case. The evaluation scenario is built upon the system designed in chapter V, during the discussion about the diverse resource allocation mechanisms. The idea is two use the winner option, rate-based demand plus random access; add the presence of some VoIP calls and apply the earliest due date differentiation scheduler presented in the previous chapter.

We have studied the average queuing delay, the jitter and the dropping rate for multiple cases raising the number of PTT groups simulated. The traffic generated by these groups follows the same principles than the one presented in chapter V. We have considered three possibilities for the voice calls: no VoIP calls, up to two calls simultaneously and up to four calls. These VoIP calls follow an ON-OFF model like the one considered in previous chapters. We have considered an exponential distribution with an average of 60 seconds to generate the call duration and with an average of 40 seconds for a new call arrival. The rest of parameters, such as symbol rate or satellite delay, are the same taken in chapter V. The examples presented here above were also generated using the same scenarios.

First, we observe the dropping rate. As we stated previously, the system works well in general, therefore, the dropping rate is very low, even in the case with no prevention, the baseline. Only when we move towards the maximum number of PTT groups generated, we see an notable increase of this metric due to the existence of more situations where the maximum capacity is exceeded. In each case, we provide the result of each service, PTT and VoIP, in every scenario (no voice calls, up to two and up to four). Figure 8.31 shows the nominal scenario. In figure figure 8.32, we observe some of the benefits of using the degraded mode for some PTT transmissions. Dropping rate is almost negligible for cases up to 30 PTT sessions and less than a half compared to the baseline for the cases with more sessions. On its turn, figure 8.33 shows a similar outcome where the start delay mode is used.

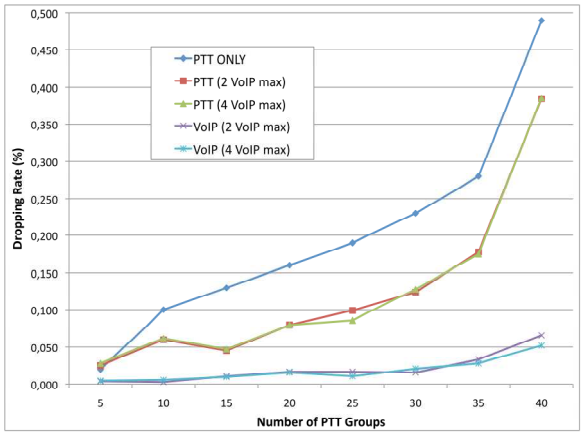


Figure 8.31: Dropping Rate. Baseline

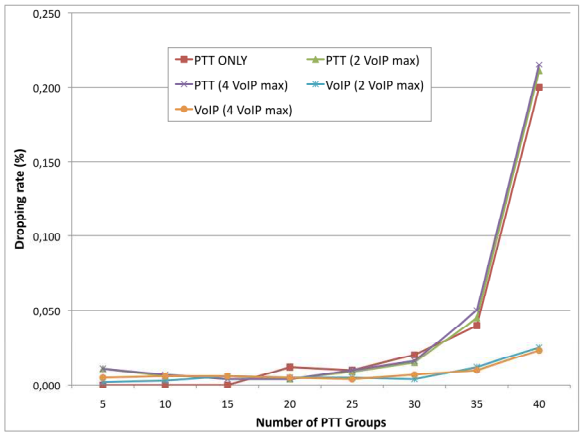


Figure 8.32: Dropping Rate. Degraded Mode

We continue by analyzing the queuing delay. Figure 8.34 provides the results for the baseline scenario. We see that the average delay is in general very low, similar to the values we observed in the

RESTRICTED DOCUMENT

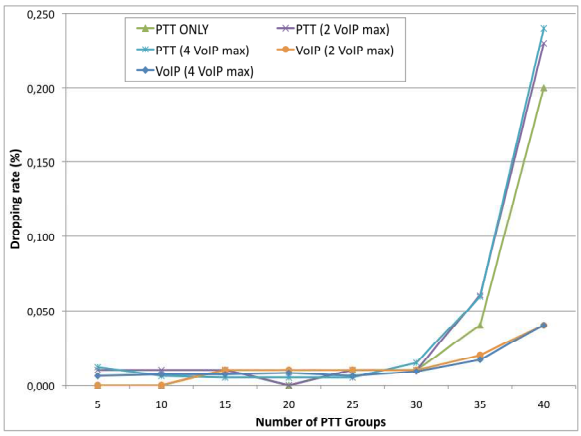


Figure 8.33: Dropping Rate. Start Delay Mode

simulations comparing the different resource demand procedures. VoIP calls are always prioritized and PTT delay increases as more sessions enter the system.

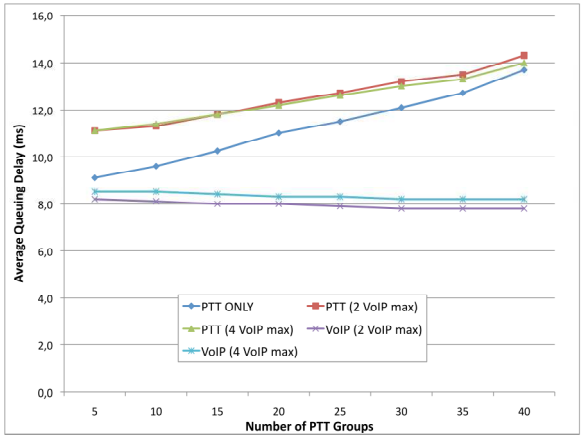


Figure 8.34: Queuing delay. Baseline

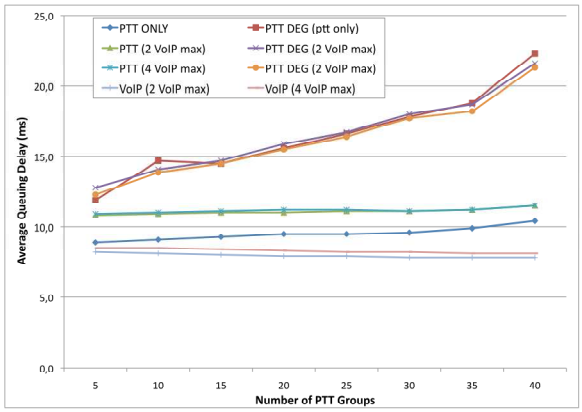


Figure 8.35: Queuing delay. Degraded Mode

In order to study the results when using the degraded mode, which can be found in figure 8.35, we provide the average delay of the PTT sessions that have been marked as degraded. For these sessions the delay increases but it is kept under acceptable levels, given that the periods where the prevention is necessary and the delay increases up to 200 *ms* are normally short. The rest of PTT sessions see their delay reduced for two reasons: when there is not an overcharge, the delay is usually low and when the degraded mode needs to be triggered the sessions that are managed under the regular method are prioritized. VoIP results are almost equal to the ones observed before.

For the start delay mode, in figure 8.36, we separate the PTT sessions that have been delayed. These sessions only occur during busy periods and consequently have a larger delay than their fellow normal PTT sessions. The latter are not prioritized like in the degraded mode. VoIP shows no appreciable change.

RESTRICTED DOCUMENT

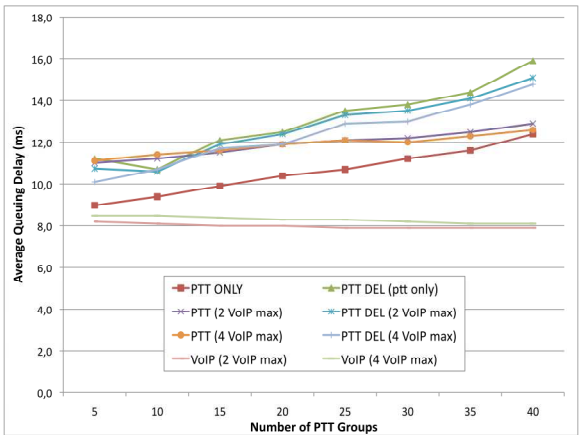


Figure 8.36: Queuing delay. Start Delay Mode

Next we review the standard deviation of the jitter that mesures the variability of the queuing delay of consecutive packets. In general we observe low values, less than 10 *ms* in most cases. Figure 8.37 provides the results when no prevention is applied. We observe a larger jitter in the sessions where the degraded mode is applied, as we see in figure 8.38. For the start delay mode, in figure 8.39, we see similar results for regular and delayed PTT sessions. This outcome is consistent because by delaying the start of some sessions, we expect to start the transmission in a less-busy period, more similar to the charge the other PTT calls experience.

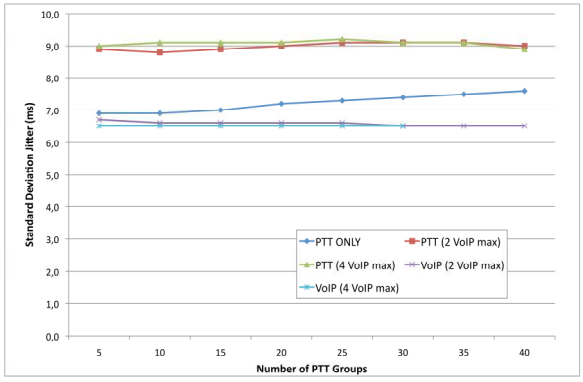


Figure 8.37: Std Jitter. Baseline

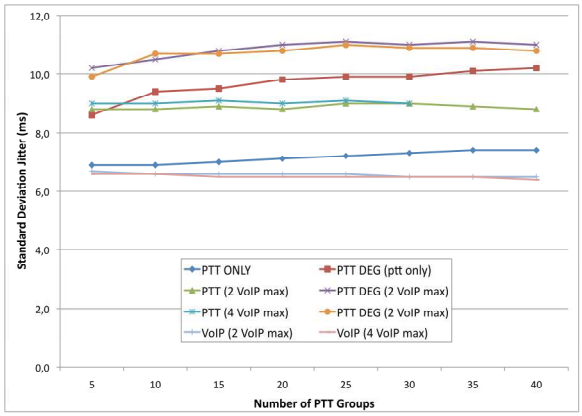


Figure 8.38: Std Jitter. Degraded Mode

Finally, we would like to observe the statistics of the start delay mode. We are interested in evaluating how long the sessions need to wait until they are started. In figure 8.40 we see that the average start delay is close to one satellite hop delay, $250 - 270$ *ms*. We indicated previously that in some cases, we could see delays up to three times those values. In order to measure the variability of the start delay duration, we include the standard deviation of it in the same figure. We observe that as the maximum number of PTT sessions increases the difference between the start delay of two sessions increases noticeably, even if the average start delay remains similar.

RESTRICTED DOCUMENT

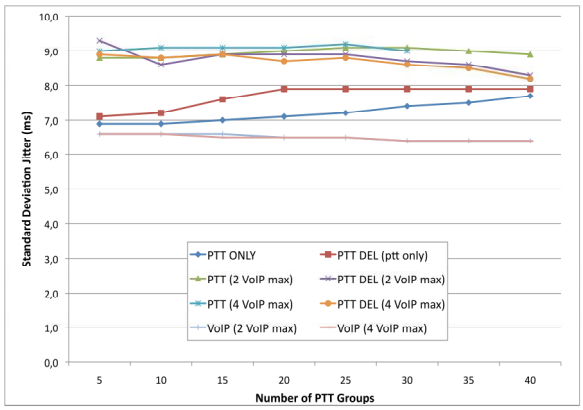


Figure 8.39: Std Jitter. Start Delay Mode

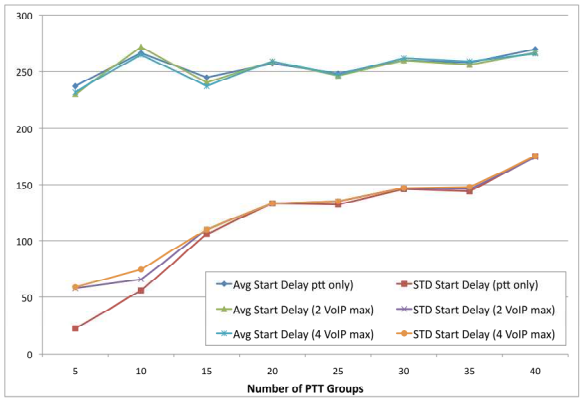


Figure 8.40: Start Delay Statistics

8.5 Conclusions

This chapter served as final study of a solution gathering multiple PTT transmissions and VoIP calls. The usage of a rate-based resource allocation and the random access mechanism keeps the key figures under good values and the usual behavior is very satisfactory. Using a differentiation procedure between PTT and VoIP helps prioritizing the latter while maintaining a good experience for the more delay tolerant PTT messages. We have observed that the major challenge is to manage the arrival of multiple sessions in a short period of time. The consequence is a larger dropping rate during those periods, whcih can result in a large burst of errors for some calls at their beginning, diminishing their quality of experience. We have proposed two preventive methods for reducing this inconvenience: a degraded mode increasing the queuing delay of some messages and the start delay mode, where the beginning of those messages is delayed waiting for periods with more rate allocation and consequently less busy.

The outcomes of both mechanisms are good and most of the bursts of errors are eliminated except for the cases where the actual maximum capacity is exceeded. However, we prefer the degraded mode because it is work conserving while the start delay mode may result in not using some allocated frames when the only packets that are available are from a message that has not yet been started. Additionally, the degradation is variable, i.e. when the busy period is over the queuing delay is similar to the ones other PTT sessions experience. Finally, the variability of the start delay duration is a negative point compared to the consistent maximum allowed delay of the degraded mode, which is known from the configuration.

RESTRICTED DOCUMENT

Chapter 9

Conclusions

The work presented in this document was motivated by the desire to conceive a Push-To-Talk solution adapted to an hybrid environment comprising satellite links and LTE mobile networks. PTT is still a key service for Public Safety agencies and other professionals relying on specific logistics and operation.

In chapter II we provided some background about the main topics treated in this thesis. Our main purpose in that part was to present the inception of PMR networks, which feature PTT as their main communication solution, and their evolution towards the LTE technology. We started first by analyzing the main users of these services, Public Safety, regarding their work during regular days and emergency situations, and their requirements in terms of telecommunications. We observed the absence of solutions specifically conceived for the satellite environment, except for the ones that tested legacy solutions with a satellite link as backhaul. Then, we reviewed the legacy PMR technologies, from no infrastructure solutions such as the analog systems or the first versions of DMR to the more complex like TETRA or P25, which are currently used in most of the countries around the world. Next, we presented LTE, its capabilities in terms of performance and the benefits regarding the usage of a full IP-based solution. Subsequently, we analyzed the advantages and opportunities for adopting LTE as the main technology for PMR networks. It is a chance to leverage a technology that is already being deployed, benefiting from the higher data rates and better interoperability. Furthermore, it presents a great opportunity for public-private-partnerships that will help governments reduce their capital expenses on network infrastructure. Our target is to provide a solution that can coexist with it and provide a key and dedicated resource for busy periods or when the main infrastructure is not available. We also evaluate the work that is currently done in this aspect within the LTE standard. Following, we discuss about Voice over IP because it relates to the same type of user data packets as in an IP-based PTT solution. Finally, we assessed the principal drawbacks of the satellite environment and we evaluate some of the techniques to overcome them. We were interested mainly in PEPs as some of the solutions we present resemble what they provide.

Then, we started the first part of this work. We discussed about the different architecture choices in order to provide a PTT solution over LTE and satellite. We reviewed the possibility of mobile satellite systems but we did not consider it a valid option because it would be necessary that all users had access to dual-mode terminals, satellite and LTE, which could not be assured. For that, we inclined ourselves

towards the deployment of temporary LTE networks. These temporary equipments could include a full LTE core or rely on an existing one on the other end of the satellite link. Finally, we concluded that the intelligence managing the PTT solution shall be provided by a set of distributed servers that we call Super Nodes. This is an in-between solution. Full distributed, i.e. all managed by the final users, is a valid approach but it could be much more complicated to manage. Further, we wanted to keep the end-user application as simple as possible. Centralized solutions are no longer valid, the presence of satellite links implies a very large disparity of latency between the central server and the users in different locations.

Chapter IV presented the distributed floor control protocol. We introduced the concept of floor control as a set of mechanisms that manage the permission to access a shared resource, which could be a file or the permission to speak. The objectives of the protocol we discuss are to overcome the issues of centralized methods, where clients closer to the moderator benefit from a lower delay and the system becomes unfair. Therefore, we focused on offering a fair solution that respects the temporal causality of the conversation and keeps a low access delay for users wishing to speak, independently of their location. We introduced the concept of an opportunistic protocol, one that allows access to an user before it is sure that he or she is the final permitted one. The consequence of this is the possible presence of conflicts that should be resolved. Our solution is based on the usage of buffering periods at every SN on a given group. Upon the reception of a request to speak, the first permission is given and the message is buffered, and after that period the message is forwarded to the listening users. During this period, the pre-granted user could change if more requests are received, the one that was issued first shall be finally accepted. We presented an analytical model in order to calculate the duration of buffer timer depending on the delay between the user and the SN, and the time between the end of previous message and the moment when the button was pushed to demand permission. It is possible that a request arrives after the end of the buffering state. In case it shall be the granted one, the system would have experienced an error, a wrong forwarding. Consequently, the first transmission would be cut in order to give access to the new arrival. Our model is based on limiting the likelihood of such events depending on the characterization of the delay and the intercall period. We presented the process of calculating the buffer period, estimating first the parameters and then computing a matrix of values from which one could retrieve the final value depending on the parameters of a certain request. We provided the state machine of a SN during the floor contention period for better understanding. We reviewed the requirements and evolution of the LTE standard to confirm that our approach fits the new specifications. In our evaluation section we observed how the length of the buffer periods varies depending on the parameters and their characterization. In the end we simulated the operation of various scenarios with two and three Super Nodes to see if the failure probability was kept under the pre-configured targets. We are satisfied with the results when the goal was 10^{-2} or 10^{-3}. However, we observed the necessity of some delay compensation in order to delay the release message of a local user to make the next contention period start almost at the same time for all users.

Afterwards, we entered into the part regarding the optimization of the user plane, mostly centered in the transmission of the PTT voice packets and the coexistence with other voice-based services such as

RESTRICTED DOCUMENT

VoIP calls. In chapter V, we examined the different resource demand methods. The capacity of the return link of a satellite system is usually shared among multiple users in order to maximize utilization. Initially, we presented the methods that are available in the DVB-RCS2, based on either rate or volume demand. We discussed about the issue that we would encounter when managing a PTT conversation. The delay difference between forward and return links would reduce the efficacy the floor control protocol. In order to assess the diverse allocation methods, we needed to generate traffic that approximates the behavior of a typical PTT group. We presented a PTT conversation model starting from the classic ON-OFF VoIP model and the results on the characterization of PTT conversations in the existing literature. Later, we reviewed some allocation methods: a committed rate throughout the duration of a conversation, a rate-based demand for each message and a combination of the latter with either a minimum allocation or the availability of a random access mechanisms. The last two yielded the best results in terms of average queuing delay and jitter thanks to the possibility to transmit frames before the feedback of a demand has been received. Yet, allocating a minimum capacity is very inefficient given that PTT groups remain non active most of time. The traffic generated by a PTT group is bursty, therefore the combination of a rate demand and random access is the best approach.

In chapter VI, we present a framework that integrates multiplexing and header compressing techniques improving the transmission efficiency of multiple voice sessions. The first mechanism gathers multiple voice payloads in a single packet whereas the second focuses on reducing the size of the packet headers by exploiting the redundancy among some of the fields in consecutive packets. Both share the same objective: reducing the percentage of the capacity dedicated to the transmission of headers. We proposed to use RoHC as header compression solution given that it has become almost the standard mechanism. We reviewed completely its design and logic. The combination of multiplexing and header compression was very promising and could result in an increase of the number of sessions served. Our simulations showed the potential to almost double this metric.

Later, in chapter VII, we focused on providing delay service differentiation between VoIP and PTT. We propose to prioritize the first service because the latter is unidirectional and could tolerate a larger delay without diminishing the quality of experience. The idea was to provide a different threshold limiting the maximum queuing delay while keeping the jitter sufficiently low and providing a similar dropping rate for both services. With this goal, we proposed three scheduling mechanisms. The first one was based on the weighted deficit round robin, which required a lot of fine tuning and had difficulties to maintain the maximum delay. For this reason we decided to take the 99th percentile as comparing metric. The same happened with one of the other mechanisms we studied, the hybrid proportional delay scheduler, which intents to perform a continuous delay differentiation given by a relative performance between services. The results were quite good and yielded the same dropping rate for VoIP and PTT. Nevertheless, it requires a large computational load because it needs to review the queuing delay of the first packet in the queue and recompute the dropping rate continuously. The last scheduler we considered was based on a single queue ordered by packet deadline. By giving each service a different deadline and removing packets when they exceed it, the scheduler did a good work limiting the delay and providing service

RESTRICTED DOCUMENT

differentiation. However, the dropping rate of both service was not consistent and further analysis should be conducted before including it in a real system.

Finally, in chapter VIII we combined the access method based on rate demands and random access with the delay differentiation scheduler based on the earliest due date in order to assess the overall simulation of multiple PTT groups and some VoIP calls. We concluded that generally the operation was satisfactory. Nevertheless, the arrival of multiple PTT messages in a short period challenged the resource allocation mode and sometimes resulted in a burst of errors, which is one of the main indications of low quality of experience, specially at the beginning of a message. To overcome this issue, we proposed two prevention methods to manage the PTT sessions that arrived in busy periods. In the degraded mode, the queuing delay for these sessions can increase up to 200 *ms* in order to prioritize the sessions already in transmission. In its turn, in the start delay mode, we propose to delay the beginning of the transmission until new resources are awarded. Both methods were adequate to reduce the presence of error bursts, but it was more difficult to evaluate the actual start delay in the latter, as it resulted quite variable from one session to another.

This thesis has resulted in two publications in international conferences [13] [15] and a patent filled in the French patent office [14]. The latter will allow Airbus Defence and Space including a novel floor control mechanism tailored for the usage of satellite links, which will definitely help them differentiate from their competitors. This work has been developed with a clear transversal vision of the Push-To-Talk service in hybrid networks, going from the control part with the floor control protocol to the user plane, considering compression and multiplexing but also the scheduling of voice-based applications.

9.1 Future Work

The work presented here in this dissertation can be extended in two directions. The first one would take advantage of the data generated during the floor control protocol operation and the second one would actually go beyond voice services and include additional services.

During the estimation of the parameters for the floor control protocol, a lot of raw data could be retrieved and processed to improve the service. We could analyze the distribution of the messages and estimate the likelihood of a super node to host the next speaker and maybe reduce the buffer duration given that the probability of receiving a request from another super node would be reduced after some events. Compared to the somewhat static approach investigated in this work, the idea here is to dynamically modulate the buffer duration based on traffic intelligence.

It could also be possible to study the distribution of the permission to speak and estimate the probability that someone from a given super node speaks after someone else has just spoken. This kind of analyses could be useful in order to provide some predictive allocation on the return link of the satellite system. This, however, would be constrained by the network control center design. Without the collaboration of the NCC, this allocation would not be possible. This pre-allocation could be less than the actual rate required for a full voice stream, but it would reduce the impact on the random channel at the

RESTRICTED DOCUMENT

beginning of new messages.

While voice is still predominant in the context of public safety, the interest in video transmission is increasing thanks to the access to high data rate capacities. Future systems could feature a push-to-video service so that in addition to the voice message from the user, a first responder would be able to share what he is seeing. During situation assessment phases, push-to-video capabilities are reported as bringing significant added value to the assessment process.

Video source coding is more complicated than voice. In the recent years, novel forms of codification have emerged. One of these techniques is known as Scalable Video Coding (SVC) [119], which allows encoding a high-quality video in multiple bit streams or layer. The lower layer sub stream carries a low-quality version of the video and the higher layers incrementally improve its quality in terms of resolution and/or frame rate.

As part of a comprehensive service, video streaming could be added to the delay service differentiation framework we presented and be transmitted together with VoIP and PTT packets. The multiple layers could receive a different treatment. The basic or lower layer would be prioritized like VoIP while higher layers could be delayed and even have a higher dropping rate in case of saturation.

Finally, also as part of a full service, we would like to introduce a new paradigm. Nowadays, telecommunication systems feature adaptive coding and modulation (ACM), which allows adapting the transmission parameters depending on the state of the propagation channel. With ACM we can increase the availability under rain fade events and improve the data rate with clear sky. We propose a similar paradigm that would operate in a higher layer.

Depending on the resource availability, we could provide an adaptive service. When a lot of capacity is available, a user could start a full duplex VoIP call; however, if resources are limited, he could still use PTT messages to communicate with his peers. Finally, if no resources were available we could deliver the message later similar to a voice mail service or vocal SMS. This last option would recall the start delay proposed mechanism but in extreme mode.

Figure 9.1 provides an overview of this adaptive service provisioning. The research would focus on finding the borders between one service and another, deciding when we need to downgrade or upgrade the provided service.

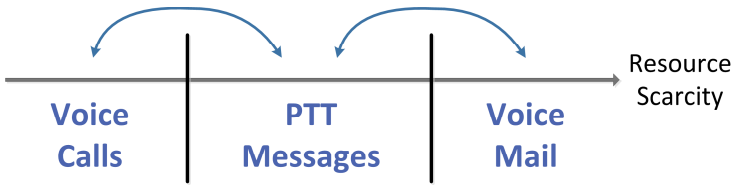


Figure 9.1: Overview of the Adaptive Service Provisioning

Finally, another aspect of adaptive service provisioning, outside the scope of this work, is the acceptance by the end-users. Indeed, compared to current practices, introducing a dynamic shift of service profile can be considered as disruptive from a Quality of Experience standpoint.

RESTRICTED DOCUMENT

Bibliography

[1] Motorola, "Long term evolution (LTE): A technical overview: White paper," 2007.

[2] P. e. a. Beming, "LTE-SAE architecture and performance," http://www.ericsson.com/ericsson/corpinfo/publications/review/2007_03/files/5_LTE_SAE.pdf, 2007, accessed: 2015-09-11.

[3] Ericsson, "Long term evolution (LTE): an introduction," http://www.mforum.ru/arc/090321_lte_overview_MForum.pdf, 2007, accessed: 2015-09-11.

[4] Agilent, "3GPP long term evolution: System overview, product development, and test challenges," http://cp.literature.agilent.com/litweb/pdf/5989-8139EN.pdf, 2007, accessed: 2015-09-11.

[5] Nokia Siemens Networks, "Voice over lte (VoLTE)," http://networks.nokia.com/fr/portfolio/solutions/voice-over-lte.

[6] Qualcomm Research, "LTE eMBMS technology overview," https://s3.amazonaws.com/sdieee/222-eMBMS_tech_overview_IEEE_112712.pdf, 2012, accessed: 2015-09-11.

[7] M. Steppler, P. Sievering, S. Kerkhoff, and T. Gray, "Evolution of TETRA," http://www.p3-group.com/downloads/4/1/7/5/P3_-_Evolution_of_TETRA_-_White_Paper_-_v1.0.pdf, 2011, accessed: 2015-09-11.

[8] Open Mobile Alliance (OMA), "Push to talk over cellular v1.0.4," http://technical.openmobilealliance.org/Technical/technical-information/release-program/current-releases/poc-v1-0-4, accessed: 2015-09-11.

[9] Y. Zhang, H. Peng, and J. Gu, "Design and implementation of a TCP performance enhancement gateway for satellite networks," in *Communications and Intelligence Information Security (ICCIIS), 2010 International Conference on*, Oct 2010, pp. 252–255.

[10] Nokia Networks, "LTE networks for public safety services," http://networks.nokia.com/file/33756/lte-for-public-safety, accessed: 2015-09-13.

[11] European Telecommunications Standards Institute (ETSI), "Second generation DVB interactive satellite system (DVB-RCS2); part 4: Guidelines for implementation and use," http://www.etsi.

org/deliver/etsi_tr/101500_101599/10154504/01.01.01_60/tr_10154504v010101p.pdf, 2014, accessed: 2015-10-08.

[12] J. Munoz, J. Ventura-Jaume, and L. Franck, "Architectures for providing public safety communications over LTE."

[13] J. Ventura-Jaume, L. Franck, and L. Girardeau, "A distributed floor control protocol for next generation PMR based on hybrid LTE and satellite networks," in *Global Humanitarian Technology Conference (GHTC), 2014 IEEE*, Oct 2014, pp. 62–69.

[14] J. Ventura-Jaume and L. Franck, "Procédé de gestion de prise de parole sur un canal de communication dans le cadre de communications en alternat," FR Patent FR 1 454 670, 5 23, 2014.

[15] J. Ventura-Jaume, L. Franck, and F. Cibaud, "Optimization of voice services on hybrid LTE and satellite networks," *21 Ka & Broadband Communications Conference*, 2015.

[16] G. Baldini, S. Karanasios, D. Allen, and F. Vergari, "Survey of wireless communication technologies for public safety," *Communications Surveys Tutorials, IEEE*, vol. 16, no. 2, pp. 619–641, Second 2014.

[17] 3GPP TS 22.179, "Mission critical push to talk (MCPTT) over LTE," http://www.3gpp.org/DynaReport/22179.htm, accessed: 2015-09-10.

[18] SAFECOM - US Department of Homeland Security, "Statement of requirements for public safety wireless communications & interoperability," http://www.emsa.ca.gov/Media/Default/PDF/sorv1.pdf, accessed: 2015-09-10.

[19] TETRA and Critical Communications Association (TCCA), "Broadband spectrum for mission critical communication needed," http://www.tandcca.com/Library/Documents/Broadband/Broadband%20Spectrum%20for%20mission%20critical%20communication%20needed_MS.pdf, accessed: 2015-09-10.

[20] Electronic Communications Comitee (ECC), "Ecc decision of 27 june 2008 on the harmonisation of frequency bands for the implementation of digital public protection and disaster relief (ppdr) radio applications in bands within the 380-470 mhz range."

[21] ——, "Ecc recommendation (08)04 the identification of frequency bands for the implementation of broad band disaster relief (bbdr) radio applications in the 5 ghz frequency range."

[22] J. M. Peha, "How america's fragmented approach to public safety wastes money and spectrum," *Telecommunications Policy*, vol. 31, 2007.

[23] US Government, "First responder network authority," http://www.firstnet.gov, accessed: 2015-09-10.

RESTRICTED DOCUMENT

[24] Airbus Defence and Space, "GSM bubble services," http://www.satcom-airbusds.com/wp-content/uploads/2014/04/gsm-bubble-datasheet.pdf, accessed: 2014-07-08.

[25] A. Via Estrem and M. Werner, "Portable satellite backhauling solution for emergency communications," in *Advanced satellite multimedia systems conference (asma) and the 11th signal processing for space communications workshop (spsc), 2010 5th*, Sept 2010, pp. 262–269.

[26] E. Fazli, M. Werner, N. Courville, M. Berioli, and V. Boussemart, "Integrated gsm/wifi backhauling over satellite: Flexible solution for emergency communications," in *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, May 2008, pp. 2962–2966.

[27] Vodafone Foundation, "Instant network," http://www.vodafone.com/content/index/about/foundation/instant-network/vodafone-instantnetwork.html, accessed: 2014-07-08.

[28] European Telecommunications Standards Institute (ETSI), "Device classes for emergency communication cells over satellite (ECCS)," http://www.etsi.org/deliver/etsi_ts/103200_103299/103284/01.01.01_60/ts_103284v010101p.pdf, accessed: 2015-09-11.

[29] D. Ruf, "TB3p base station connection to a DXT3 via satellite link - test report," 2012.

[30] A. Donner, J. A. Saleemi, and J. Mulero Chaves, "Backhauling TETRA via satellite networks," *International Journal of Communications Systems*, vol. 1-6, 2000.

[31] P. D. Karabinis, "Satellite assisted push-to-send radio terminal systems and methods," EU Patent EP2 209 222 A3, 10 06, 2010.

[32] European Telecommunications Standards Institute (ETSI), "Digital mobile radio," http://www.etsi.org/technologies-clusters/technologies/digital-mobile-radio, accessed: 2015-09-11.

[33] ——, "Terrestrial trunked radio (TETRA)," http://www.etsi.org/technologies-clusters/technologies/tetra, accessed: 2015-09-11.

[34] ——, "TETRA enhanced data service (TEDS)," http://www.etsi.org/deliver/etsi_tr/102500_102599/102580/01.01.01_60/tr_102580v010101p.pdf, accessed: 2015-09-11.

[35] TETRAPOL Forum, "TETRAPOL," http://www.tetrapol.com, accessed: 2015-09-11.

[36] GSM Association, "IMS profile for voice and SMS," http://www.gsma.com/newsroom/wp-content/uploads/IR.92-v9.0.pdf, 2007, accessed: 2015-09-11.

[37] L. Cheng, X. Tong, J. Wu, and B. Kong, "Analysis and simulation of HPAs' effects on satellite OFDM systems," in *Information Science and Technology (ICIST), 2013 International Conference on*, March 2013, pp. 1212–1216.

RESTRICTED DOCUMENT

[38] S. Janaaththanan, C. Kasparis, and B. Evans, "Feasibility study of adaptive lut-based pre-distorter for OFDM in non-linear satellite downlink channel," in *Satellite and Space Communications, 2006 International Workshop on*, Sept 2006, pp. 126–129.

[39] A. T. Ho, M.-L. Boucheret, N. Thomas, M. Dervin, and X. Deplancq, "OFDM synchronization scheme for the forward link of a fixed broadband satellite system," in *Signal Processing Advances in Wireless Communications, 2008. SPAWC 2008. IEEE 9th Workshop on*, July 2008, pp. 336–340.

[40] Z. Deng, H. Wang, M. Tang, X. Gao, and X. You, "A new PSS design and identification for global coverage multi-beam S-LTE using generalized zadoff-chu sequences," in *Wireless Communications Signal Processing (WCSP), 2013 International Conference on*, Oct 2013, pp. 1–5.

[41] V. Jungnickel, H. Gaebler, U. Krueger, K. Manolakis, and T. Haustein, "LTE trials in the return channel over satellite," in *Advanced Satellite Multimedia Systems Conference (ASMS) and 12th Signal Processing for Space Communications Workshop (SPSC), 2012 6th*, Sept 2012, pp. 238–245.

[42] Wikipedia, "Lightsquared," https://en.wikipedia.org/wiki/LightSquared, accessed: 2015-09-11.

[43] M. Crosnier, F. Planchou, R. Dhaou, and A. Beylot, "Handover management optimization for LTE terrestrial network with satellite backhaul," in *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, May 2011, pp. 1–5.

[44] M. Crosnier, R. Dhaou, F. Planchou, and A. Beylot, "TCP performance optimization for handover management for LTE satellite/terrestrial hybrid networks," in *Satellite Telecommunications (ESTEL), 2012 IEEE First AESS European Conference on*, Oct 2012, pp. 1–5.

[45] M. Breiling, W. Zia, Y. Sanchez de la Fuente, V. Mignone, D. Milanesio, Y. Fan, and M. Guta, "LTE backhauling over MEO-satellites," in *Advanced Satellite Multimedia Systems Conference and the 13th Signal Processing for Space Communications Workshop (ASMS/SPSC), 2014 7th*, Sept 2014, pp. 174–181.

[46] Verizon Wireless, "The verizon wireless 4G LTE network: Transforming business with next-generation technology," https://business.verizonwireless.com/content/dam/b2b/resources/LTE_FutureMobileTech_WP.pdf, accessed: 2015-09-11.

[47] R. Ferrus, O. Sallent, G. Baldini, and L. Goratti, "LTE: the technology driver for future public safety communications," *Communications Magazine, IEEE*, vol. 51, no. 10, pp. 154–161, October 2013.

[48] National Public Safety Telecommunications Council, "Public safety broadband: Push-to-talk over long term evolution requirements," http://www.npstc.org/download.jsp?tableId=37&column=217&id=2813&file=PTT_Over_LTE_Master_130719.pdf, 2013, accessed: 2015-09-11.

RESTRICTED DOCUMENT

[49] S. K. Das, "Feasibility study of ip multimedia subsystem (IMS) based push-to-talk over cellular (PoC) for public safety and security communications," Master's thesis, Helsinki University of Technology, 2006.

[50] C. Lu, "Delay analysis of push-to-talk over cellular (PoC) service solutions for public safety communications over lte networks," Master's thesis, Technical University of Catalonia, 2012.

[51] 3GPP TS 23.468, "Group communication system enablers for LTE (GCSE-LTE)," http://www.3gpp.org/DynaReport/23468.htm, accessed: 2015-09-11.

[52] 3GPP TS 22.803, "Feasibility study for proximity services (ProSe)," http://www.3gpp.org/DynaReport/22803.htm, accessed: 2015-09-11.

[53] 3GPP TS 22.346, "Isolated evolved universal terrestrial radio access network (E-UTRAN) operation for public safety," http://www.3gpp.org/DynaReport/22346.htm, accessed: 2015-09-11.

[54] 3GPP TS 22.278, "Service requirements for the evolved packet system (EPS)," http://www.3gpp.org/DynaReport/22278.htm, accessed: 2015-09-11.

[55] International Telecommunications Union, "ITU-T G series: Transmission systems and media, digital systems and networks," http://www.itu.int/net/itu-t/sigdb/speaudio/Gseries.htm, accessed: 2015-09-11.

[56] A. Ramo, "Voice quality evaluation of various codecs," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, March 2010, pp. 4662–4665.

[57] S. Oueslati-Boulahia, A. Serhrouchni, S. Tohme, S. Baier, and M. Berrada, "TCP over satellite links: Problems and solutions," *Telecommunications Systems*, vol. 13, no. 2-4, pp. 199–212, July 2000.

[58] Internet Engineering Task Force (IETF), "Request for comments: 6582. the newreno modification to TCP's fast recovery algorithm," http://www.itu.int/net/itu-t/sigdb/speaudio/Gseries.htm, 2012, accessed: 2015-09-11.

[59] C. Caini and R. Firrincieli, "TCP hybla: a TCP enhancement for heterogeneous networks," *International Journal of Satellite Communications and Networking*, vol. 22, pp. 547–566, 2004.

[60] J. Shen, H. Yu, X. Zhang, and D. Liao, "A performance enhancing proxy for terrestrial-satellite hybrid networks," in *Communications, Circuits and Systems, 2008. ICCCAS 2008. International Conference on*, May 2008, pp. 529–533.

[61] T. T. Thai, D. Pacheco, E. Lochin, and F. Arnal, "SatERN: A PEP-less solution for satellite communications," in *Communications (ICC), 2011 IEEE International Conference on*, June 2011, pp. 1–5.

RESTRICTED DOCUMENT

[62] C. Caini, R. Firrincieli, and D. Lacamera, "PEPsal: a performance enhancing proxy designed for TCP satellite connections," in *Vehicular Technology Conference, 2006. VTC 2006-Spring. IEEE 63rd*, vol. 6, May 2006, pp. 2607–2611.

[63] P. Woodward, T. Pell, and G. Hernandez, "Performance enhancing proxies - are they as beneficial as they seem?" 2003.

[64] N. Ehsan, M. Liu, and J. R. Roderick, "Evaluation of performance enhancing proxies in internet over satellite," *International Journal of Communications Systems*, January 2003.

[65] J. Ishac and M. Allman, "On the performance of TCP spoofing in satellite networks," in *Military Communications Conference, 2001. MILCOM 2001. Communications for Network-Centric Operations: Creating the Information Force. IEEE*, vol. 1, 2001, pp. 700–704 vol.1.

[66] E. Dubois, J. Fasson, C. Donny, and E. Chaput, "Enhancing TCP based communications in mobile satellite scenarios: TCP PEPs issues and solutions," in *Advanced satellite multimedia systems conference (asma) and the 11th signal processing for space communications workshop (spsc), 2010 5th*, Sept 2010, pp. 476–483.

[67] Thuraya, "Thuraya satleeve for iphone or android," http://www.thuraya.com/SatSleeve, accessed: 2015-11-24.

[68] H.-P. Dommel and J. Garcia-Luna-Aceves, "Floor control for multimedia conferencing and collaboration," 1997.

[69] M. Al Rubaye, F. Belqasmi, C. Fu, and R. Glitho, "A novel architecture for floor control in the IP multimedia subsystem of 3G networks," in *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th*, 2009, pp. 1–5.

[70] J. Garcia-Luna-Aceves, P. Mantey, and S. Potireddy, "Floor control alternatives for distributed videoconferencing over IP networks," in *Collaborative Computing: Networking, Applications and Worksharing, 2005 International Conference on*, 2005, pp. 10 pp.–.

[71] Q. Wang, H. Jiang, A. Wong, J. Li, and Z. Li, "A full-distributed architecture for PoC application in data packet voice communication," in *Computer and Information Technology (CIT), 2011 IEEE 11th International Conference on*, 2011, pp. 231–237.

[72] S. Banik, S. Radhakrishnan, V. Sarangan, and C. Sekharan, "Implementation of distributed floor control protocols on overlay networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 19, no. 8, pp. 1057–1070, 2008.

[73] K. Kostas Katrinis, G. Parissidis, and B. Plattner, "Activity sensing floor control in multimedia collaborative applications," in *Proc. of the 10th International Conference on Distributed Multimedia Systems (DMS)*, 2004.

RESTRICTED DOCUMENT

[74] E. Bouwers, "Fast inter system push to talk operation," EU Patent EP 2 160 050B1, 11 21, 2012.

[75] S. Shaffer et al., "Floor control over high latency networks in an interoperability and collaboration system," US Patent US 20 090 054 010 A1, 02 26, 2009.

[76] A. D. Abbate et al., "Floor control over high latency networks in an interoperability and collaboration system," US Patent US 8 032 169 B2, 05 28, 2009.

[77] T. D. Bekiares et al., "Floor control in a communication system," US Patent US 8 531 993 B2, 04 30, 2009.

[78] K. Balachandran et al., "Adaptive method of floor control with fast response time and fairness in communication network," US Patent US 7 873 067 B2, 01 18, 2011.

[79] C.-H. Gan and Y.-B. Lin, "Push-to-talk service for intelligent transportation systems," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 8, no. 3, pp. 391–399, 2007.

[80] N. Aschenbruck, P. Martini, and M. Gerharz, "Characterisation and modelling of voice traffic in first responder networks," in *Local Computer Networks, 2007. LCN 2007. 32nd IEEE Conference on*, Oct 2007, pp. 295–302.

[81] J. Hernandez and I. Phillips, "Weibull mixture model to characterize end-to-end internet delay at coarse time-scales," *Communications, IEE Proceedings-*, vol. 153, no. 2, pp. 295–304, April 2006.

[82] L. Carla, R. Fantacci, F. Gei, D. Marabissi, and L. Micciullo, "LTE enhancements for public safety and security communications to support group multimedia communications," *CoRR*, vol. abs/1501.03613, 2015.

[83] Nsnam, "NS-3 network simulator," https://www.nsnam.org, accessed: 2015-09-11.

[84] Fraunhofer Institute, "Fraunhofer on-board processor (FOBP)," http://www.iis.fraunhofer.de/content/dam/iis/en/doc/ks/hfs/FOBP_Flyer_Englisch.pdf, accessed: 2015-11-24.

[85] P. Mitchell, D. Grace, and T. Tozer, "Adaptive burst targeted demand assignment multiple access (BTDAMA) for geostationary satellite systems," in *Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004. 15th IEEE International Symposium on*, vol. 4, Sept 2004, pp. 2489–2493 Vol.4.

[86] X. Yuanbo, Z. Xi, S. Yang, and P. Yaohua, "Performance evaluation of the HRDAMA-PB MAC protocol for GEO satellite networks," in *Computer Research and Development (ICCRD), 2011 3rd International Conference on*, vol. 4, March 2011, pp. 41–44.

[87] V. Leung, "Mobile radio group communications by satellite," *Vehicular Technology, IEEE Transactions on*, vol. 42, no. 2, pp. 121–130, May 1993.

RESTRICTED DOCUMENT

[88] L. G. Roberts, "ALOHA packet systems with and without slots and capture," *ARPANET System Note 8*, June 1972.

[89] L. Choudhury Gagan and S. Rappaport Stephen, "Diversity ALOHA - a random access scheme for satellite communications," *Communications, IEEE Transactions on*, vol. 31, no. 3, pp. 450–457, Mar 1983.

[90] O. Herrero, G. Foti, and G. Gallinaro, "Spread-spectrum techniques for the provision of packet access on the reverse link of next-generation broadband multimedia satellite systems," *Selected Areas in Communications, IEEE Journal on*, vol. 22, no. 3, pp. 574–583, April 2004.

[91] E. Casini, R. De Gaudenzi, and O. Herrero, "Contention resolution diversity slotted ALOHA (CRDSA): An enhanced random access schemefor satellite access packet networks," *Wireless Communications, IEEE Transactions on*, vol. 6, no. 4, pp. 1408–1419, April 2007.

[92] International Telecommunications Union - Telecommunication Standardization Sector, "P.59: Artificial conversational speech," https://www.itu.int/rec/T-REC-P.59-199303-I/en, 1993, accessed: 2015-10-08.

[93] P. T. Brady, "A technique for investigating on-off patterns of speech," *Bell System Technical Journal, The*, vol. 44, no. 1, pp. 1–22, Jan 1965.

[94] G. Hess and J. Cohn, "Communication load and delay in mobile trunked systems," in *Vehicular Technology Conference, 1981. 31st IEEE*, vol. 31, April 1981, pp. 269–273.

[95] P. Cohen, D. Haccoun, and H. H. Hoc, "Traffic analysis for different classes of users of land mobile communication systems," in *Vehicular Technology Conference, 1983. 33rd IEEE*, vol. 33, May 1983, pp. 283–285.

[96] F. Barcelo and S. Bueno, "Idle and inter-arrival time statistics in public access mobile radio (PAMR) systems," in *Global Telecommunications Conference, 1997. GLOBECOM '97., IEEE*, vol. 1, Nov 1997, pp. 126–130 vol.1.

[97] D. Sharp, N. Cackov, N. Laskovic, Q. Shao, and L. Trajkovic, "Analysis of public safety traffic on trunked land mobile radio systems," *Selected Areas in Communications, IEEE Journal on*, vol. 22, no. 7, pp. 1197–1205, Sept 2004.

[98] K. Sriram and W. Whitt, "Characterizing superposition arrival processes in packet multiplexers for voice and data," *Selected Areas in Communications, IEEE Journal on*, vol. 4, no. 6, pp. 833–846, Sep 1986.

[99] H. Sze, S. Liew, J. Lee, and D. Yip, "A multiplexing scheme for H.323 voice-over-IP applications," *Selected Areas in Communications, IEEE Journal on*, vol. 20, no. 7, pp. 1360–1368, Sep 2002.

RESTRICTED DOCUMENT

[100] M. Abu-Alhaj, M. Kolhar, L. Chandra, O. Abouabdalla, and A. Manasrah, "Delta-multiplexing: A novel technique to improve VoIP bandwidth utilization between VoIP gateways," in *Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on*, June 2010, pp. 329–335.

[101] M. Abu-Alhaj, M. Kolhar, M. Halaiyqah, O. Abouabdalla, and R. Sureswaran, "Muxcomp - a new architecture to improve VoIP bandwidth utilization," in *Future Networks, 2009 International Conference on*, March 2009, pp. 212–215.

[102] W. Wang, S. Liew, Q. Pang, and V. Li, "A multiplex-multicast scheme that improves system capacity of voice-over-IP on wireless LAN by 100%," in *Computers and Communications, 2004. Proceedings. ISCC 2004. Ninth International Symposium on*, vol. 1, June 2004, pp. 472–477 Vol.1.

[103] A. Drozdy, A. Rakos, Z. Vincze, and C. Vulkan, "Adaptive VoIP multiplexing in LTE backhaul," in *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, May 2011, pp. 1–6.

[104] S. Casner and V. Jacobson, "IETF RFC 2508. compressing IP/UDP/RTP headers for low-speed serial links," https://tools.ietf.org/html/rfc2508, 1999, accessed: 2015-09-11.

[105] T. Koren, S. Casner, J. Geevarghese, B. Thompson, and P. Ruddy, "IETF RFC 3545. enhanced compressed RTP (CRTP) for links with high delay, packet loss and reordering," https://tools.ietf.org/html/rfc3545, 2003, accessed: 2015-09-11.

[106] B. Degermark, M.and Nordgren and S. Pink, "IETF RFC 2507. IP header compression," https://tools.ietf.org/html/rfc2507, 1999, accessed: 2015-09-11.

[107] C. e. a. Bormann, "IETF RFC 3095. robust header compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed," https://tools.ietf.org/html/rfc3095, 2001, accessed: 2015-09-11.

[108] G. e. a. Pelletier, "IETF RFC 5225. robust header compression version 2 (ROHCv2): Profiles for RTP, UDP, IP, ESP and UDP-lite," https://tools.ietf.org/html/rfc5225, 2001, accessed: 2015-09-11.

[109] M. Anehill, "Validating voice over LTE end-to-end," *Ericsson Review*, 2012.

[110] M. Shreedhar and G. Varghese, "Efficient fair queuing using deficit round-robin," *Networking, IEEE/ACM Transactions on*, vol. 4, no. 3, pp. 375–385, Jun 1996.

[111] C. Dovrolis, D. Stiliadis, and P. Ramanathan, "Proportional differentiated services: delay differentiation and packet scheduling," *Networking, IEEE/ACM Transactions on*, vol. 10, no. 1, pp. 12–26, Feb 2002.

RESTRICTED DOCUMENT

[112] C.-C. Li, S.-L. Tsao, M. C. Chen, Y. Sun, and Y.-M. Huang, "Proportional delay differentiation service based on weighted fair queuing," in *Computer Communications and Networks, 2000. Proceedings. Ninth International Conference on*, 2000, pp. 418–423.

[113] H.-T. Ngin and C.-K. Tham, "Achieving proportional delay differentiation efficiently," in *Networks, 2002. ICON 2002. 10th IEEE International Conference on*, 2002, pp. 169–174.

[114] C.-C. Wu, H.-M. Wu, C.-L. Liu, and W. Lin, "A DDRR-based scheduler to achieve proportional delay differentiation in terabit network," in *Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference on*, vol. 2, March 2005, pp. 347–350 vol.2.

[115] D. Ippoliti, X. Zhou, and L. Zhang, "Packet scheduling with buffer management for fair bandwidth sharing and delay differentiation," in *Computer Communications and Networks, 2007. ICCCN 2007. Proceedings of 16th International Conference on*, Aug 2007, pp. 569–574.

[116] M. Shreedhar and G. Varghese, "Efficient fair queuing using deficit round-robin," *Networking, IEEE/ACM Transactions on*, vol. 4, no. 3, pp. 375–385, Jun 1996.

[117] S. Bodamer, "A scheduling algorithm for relative delay differentiation," in *High Performance Switching and Routing, 2000. ATM 2000. Proceedings of the IEEE Conference on*, 2000, pp. 357–364.

[118] C. Dovrolis and P. Ramanathann, "Proportional differentiated services, part ii: loss rate differentiation and packet dropping," in *Quality of Service, 2000. IWQOS. 2000 Eighth International Workshop on*, 2000, pp. 53–61.

[119] 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, Sept 2007.

RESTRICTED DOCUMENT

Résumé

Le Push-To-Talk (PTT) est un service de communication simple qui permet à un groupe de personnes d'échanger des messages vocaux. Le PTT est très populaire parmi les forces de l'ordre et d'autres services d'urgence. Sa nature unidirectionnelle et son fonctionnement réglementé permet aux utilisateurs de parler sans interruption de façon ordonné.

Les systèmes de radiocommunications mobiles professionnelles (PMR) fournissent des services voix et donnés aux intervenants des premier secours et maintiennent le PTT comme leur fonction la plus utilisée. Les solutions PMR de la prochaine génération sont susceptibles de converger avec les technologies mobiles terrestres publiques telles que Long Term Evolution (LTE) afin de bénéficier de débits plus élevés et d'une meilleure interopérabilité avec d'autres réseaux existants. Lors d'événements catastrophiques, les infrastructures de télécommunications pourraient être détruites, saturées ou absentes. Par conséquent, il est utile de déployer des réseaux temporaires sur les régions isolées et de les connecter au réseau de cœur à travers de satellites géostationnaires. Dans cette thèse, nous étudions l'adaptation de l'application PTT à cet environnement hybride basé sur des liaisons par satellite et les réseaux LTE. Les solutions existantes ont été conçues sans tenir compte du satellite comme un élément clé du système. Par conséquent, nous saisissons l'occasion de l'évolution des systèmes PMR pour intégrer les défis du cadre des liens satellitaires.

Mots-clés : Push-To-Talk, Satellite, Long Term Evolution, Professional Mobile Radio, Qualité de Service, Sécurité Civile

Abstract

Push-To-Talk (PTT) is a simple communication service that allows a group of people to exchange voice messages. PTT is very popular among law enforcement agents and other emergency services. Its half-duplex nature and regulated operation helps users speak without interruption in an ordered fashion.

Professional Mobile Radio (PMR) systems provide voice and data services to first responders and maintain PTT as their most used feature. Next generation PMR solutions are likely to converge with public land mobile technologies such as Long Term Evolution (LTE) to benefit from higher data rates and better interoperation with other existing networks. During catastrophic events, the supporting infrastructure networks could be destroyed, saturated or absent. Therefore, it is helpful to deploy temporary networks on the isolated areas and backhaul them by means of geostationary satellites to connect to the core backbone. In this thesis, we study the adaptation of the PTT application to this hybrid environment based on satellite links and LTE networks. Legacy solutions were designed without considering the satellite as a key part of the system. Hence, we take the opportunity of the PMR systems evolution to incorporate the challenges of the satellite framework.

Keywords : Push-To-Talk, Satellite, Long Term Evolution, Professional Mobile Radio, Quality of Service, Public Safety

TELECOM Bretagne

n° d'ordre : 2016telb0398

Télécom Bretagne

Technopôle Brest-Iroise - CS 83818 - 29238 Brest Cedex 3

Tél : + 33(0) 29 00 11 11 - Fax : + 33(0) 29 00 10 00